НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ
ІНСТИТУТ імені ІГОРЯ
СІКОРСЬКОГО»

факультет інформатики та обчислювальної техніки
(повна назва інституту/факультету)

кафедра автоматика та управління в технічних системах
(повна назва кафедри)

«На правах рукопису»
УДК_____

«До захисту допущено»
Завідувач кафедри
_____ О. І. Ролік
(підпис) (ініціали, прізвище)
“___”_____2019 р.

Магістерська дисертація

зі спеціальності (спеціалізації) 126 «Інформаційні системи та технології»
(код і назва спеціальності)

на тему: Підсистема аутентифікації даних користувача в інформаційно-телекомунікаційній системі

Виконав : студент 6 курсу, групи ІА–82мп
(шифр групи)

Чернишевич Богдана Сергіївна _____ _____
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник доцент, к.т.н. Полторак В.П. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____ _____ _____
(назва розділу) (науковий ступінь, вчене звання, , прізвище, ініціали) (підпис)

Рецензент _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)(підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2019 року

АНОТАЦІЯ

В даній роботі 103 сторінки текстової інформації, 20 рисунків та 40 таблиці.

Актуальність даної роботи полягає в наявності проблеми аутентифікації даних в інформаційно-телекомунікаційних системах, оскілки в зв’язку з постійним ростом продуктивності обчислювальних засобів необхідно постійно удосконалювати існуючі криптографічні схеми для збереження необхідного рівня стійкості.

Існує два рішення даної проблеми. Найпростіший спосіб – це використання ключів більшої довжини, що призводить до зростання часу роботи алгоритму аутентифікації.

Або ж використання алгоритмів заснованих на проблемі вирішення більш важкої задачі.

Метою магістерської дисертації є створення автоматизованої підсистеми аутентифікації даних зі зменшеною довжиною ключів та зі збереженням високого рівня стійкості.

Об’єктом є підсистема аутентифікації даних користувачів

Предметом є алгоритми створення пари ключів для підписання та перевірки підписаних даних користувачів

Ключові слова: підсистеми аутентифікації, некомутативні групи, розпаралелення розхрахунків, автоматизація ЕЦП.

ANNOTATION

In this paper 103 pages of textual information, 20 figures and 40 tables.

The relevance of this work is the problem of authentication of data to information and telecommunication systems, fragments due to the constant increase in the productivity of computing facilities need to constantly improve existing cryptographic schemes to maintain the required level of stability.

There are two solutions to this problem. The simplest way is to use longer-length keys to produce an authentication algorithm for hours to grow.

Or the use of algorithms Based on the problem of solving a more difficult problem.

The purpose of the master's thesis is to create an Automated Data Authentication subsystem by Reducing the length of keys and maintaining a high level of stability.

The object is the User Data Authentication Subsystem

The subject is algorithms for creating a key pair for Signing and Verification of signed User Data

Keywords: authentication subsystems, non-commutative groups, parallelization of calculations, EDS automation.

ЗМІСТ

ВСТУП

Інформаційно-телекомунікаційна система - це система передачі даних, призначена переважно для передачі інформації, доступ до якої здійснюється з використанням комп'ютерних та інших технічних пристроїв.

На даний момент можливості інформаційно-телекомунікаційних систем практично необмежені, так як передавати інформацію можна великою кількістю способів: відеозаписи, аудіозаписи, документи, графіка.

Велика кількість способів передачі інформації, а також висока точність і швидкість передачі даних можлива завдяки високому рівню розвитку інформаційно-телекомунікаційних систем.

У сучасних умовах розвитку інформаційних систем і технологій, кількість електронних даних безперервно зростає. З кожним днем збільшення кількості інформації, яка зберігається, обробляється і при необхідності передається. Це призводить до підвищення актуальності проблеми забезпечення комплексного захисту данних користувачів: забезпечення конфіденційності (неможливість отримання даних третіми особами), цілісності (запобігання несанкціоннірованого зміни інформації) даних; а також підтвердження автентичності даних, і при необхідності підтвердження прав авторства користувача.

Актуальність теми. Проблема аутентифікації даних користувачів в інформаційно-телекомунікаційних системах є одним з ключових аспектів інформаційної безпеки.

Аутентификация даних – підтвердження того, що дані належать первному користувачу, який являється їхнім атором, а також підтвердження справжності змісту повідомлення, часу генерації повідомлення і т.д.

Для вирішення завдання аутентифікації даних використовується концепція елетронно-цифрового підпису. Під терсіном електронно-цифровий підпис мається на увазі методи, що дозволяють встановлювати справжність автора даних у разі виникнення сумнівів щодо авторства.

Головні елементи електронного підпису - ключі (відкритий, тобто направляється адресату, і закритий, доступ до якого є тільки у людини, яка отримала ЦП). Перший передається разом з спрямованим файлом (даними) отримувачу, другий залишається у відправника.

Сьогодні широко використовуються схеми підписів, заснованих на асиметричних криптографічних алгоритмах. Вони дозволяють отримати найбільш захищені від підробки реквізитів.

Наразі найчастіше для побудови схем електронно-цифрового підпису використовується алгоритм RSA. В основі даного алгоритму лежить концепція протоколу обміну ключами Діффі-Хеллмана. Сутність цього протоколу полягає у тому, що кожен користувач мережі має пару ключів: закритий ключ, який необхідний для формування підпису та відкритий ключ, призначений для перевірки підпису, який може бути відомим іншим користувачам мережі.

Існує велика кількість алгоритмів і протоколів аутентифікації даних, побудованих на різних криптографічних схемах, які використовуються для аутентифікації даних користувача.

У зв'язку з постійним зростанням продуктивності обчислювальних засобів необхідно постійно вдосконалювати існуючі криптосхеми. Удосконалення криптосистем дозволить зберігати необхідний рівень стійкості, або ж збільшити рівень криптостійкості.

Останнім часом великий інтерес представляють криптосхеми, засновані на вирішенні нових важких задач, наприклад завдання прихованого дискретного логарифмирования. Вона об'єднує в собі задачу пошуку сопрягающего елемента і дискретного логарифмування в прихованій комутативній підгрупі, які є перспективними для побудови протоколів відкритого узгодження секретного ключа і відкритого шифрування підвищеної криптостойкості.

Однак, криптосхеми, побудовані з використанням завдання прихованого дискретного логарифмування, мають серйозний недолік – низьку продуктивність.

Цей недолік обумовлений необхідністю використання в таких кріптосхеми

некомутативних груп, операції в яких вимагають великих обчислювальних витрат. У зв'язку з цим є проблема підвищення продуктивності криптосхему, заснованих на некомутативних кінцевих групах.

Метою дисертації є створення підсистеми аутентифікації з підвищеною криптостійкістю зі збереженням продуктивності алгоритму.

Задачі:

- Дослідження алгоритму некомутативного шифрування для процесу аутентификації
- Реалізація алгоритму електроно-цифрового підпису зі зменшеною довжиною ключів
- Підвищення продуктивності обраного методу
- Реалізація підсистеми для аутентифікації даних в автоматизованій інформаційно-телекомунікаційній системі

Об’єктом є підсистема аутентифікації даних користувачів

Предметом є алгоритми створення пари ключів для підписання та перевірки підписаних даних користувачів

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

Для аутентіфікації даних на даний момент існує достатньо велика кількість алгоритмів та протоколі, які базуються на різних математичних моделях та криптографічних схемах.

Під аутентифікацією даних в інформаційно-телекомунікаційних системах мається на увазі встановлення достовірності даних, отриманих по мережі, виключно на основі інформації, що міститься в отриманих даних.

Якщо кінцевою метою шифрування даних є забезпечення захисту від несанкціонованого ознайомлення з цією інформацією, то кінцевою метою аутентифікації інформації є забезпечення захисту учасників інформаційного обміну від нав'язування хибної інформації.

Концепція аутентифікації в широкому сенсі передбачає встановлення достовірності даних як за умови наявності взаємної довіри між учасниками обміну, так і при його відсутності.

У даного розділі описані базові алгоритми електронно-цифрового підпису, а також аналіз рішень, які існують на даний момент.

1.1 Алгоритми електронно-цифрового підпису

Для вирішення завдання аутентифікації інформації використовується концепція електронно-цифрового підпису. Електронно-цифровий підпис є методом, що дозволяє встановлювати справжність автора даних при виникненні спору щодо авторства цих даних.

Використання електронно-цифрового підпису дозволяє здійснити:

- Контроль цілісностності переданих даних: при будь-якому випадковому або навмисному зміні даних підпис стане недійсним, так як вона була обчислена на підставі вихідного стану і відповідає лише йому.

- Захист від змін (підробки) даних: гарантія виявлення підробки при контролі цілісності робить підроблення недоцільним в більшості випадків.
- Неможливість відмови від авторства. Так як для створення коректного підпису необхідно знати закритий ключ, а він повинен бути відомий тільки власнику, то власник не може відмовитися від свого підпису під даними.

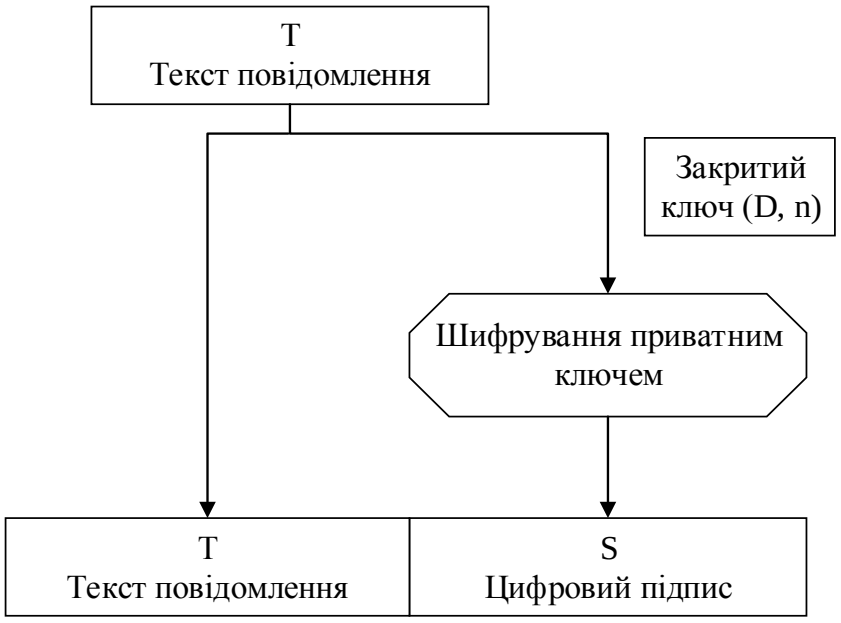

Рисунок 1.1 – Схема використання ЕЦП

Симетричні схеми. Такі ЕЦП менш поширені ніж асиметричні, так як після появи концепції цифрового підпису не вдалося реалізувати ефективні алгоритми підпису, засновані на відомих на той час симетричних шифрах. Дана схема передбачає наявність у системі третьої особи - особи, що користується довірою обох сторін. Підтвердженням достовірності даних є сам факт шифрування цих данних секретним ключем і передача їх довіреній особі.

Реалізація електронно-цифрового підпису за допомогою симетричної схеми полягає в створенні якогось центрального авторитетного органу, якому всі довіряють, - довіреної особи. Потім кожен користувач обирає секретний ключ і особисто відносить його довіреній особі. Таким чином, секретний

ключ K_A Аліси відомий тільки Алісі і довіреній особі.

Коли Аліса хоче послати відкритим текстом Бобу підписане повідомлення P, вона формує повідомлення, зашифровує його своїм секретним ключем $K_A(B, R_A, t,\ P)$повідомлення з ідентифікатором B, випадковим числом R_A, временим штампом t. Потім Аліса посилає це повідомлення довіреній особі. Довірена особа бачить, що це повідомлення від Аліси, розшифровує його і посилає Бобу. Повідомлення, яке отримає Боб, буде містити відкритий текст повідомлення Аліси і підпис довіреної особи.

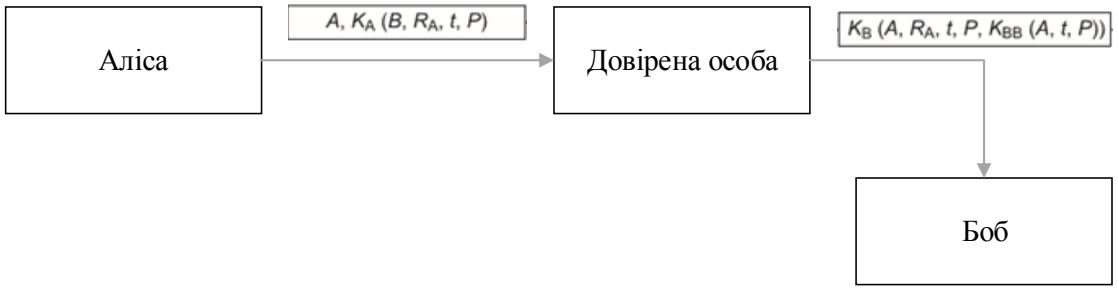

Рисунок 1.2 – Алгоритм роботи симетричного ЕЦП

Переваги симетричної схеми ЕЦП:

- Стійкість симетричного ЕЦП базуеться на стійкості блочного шифру, який він використовує.
- Якщо стійкість шифру буде недостатньо висока, його легко можно замінити на більш стійкий з мінімальними змінами

Недоліки симетричної схеми ЕЦП:

- Необхідно окремо підписувати кожний біт переданої інформації, що призводить до значного збільшення розміру підпису
- Сгенеровані для підпису ключі можуть використані лише один раз, так як під час підписання розкривается половина серкетного ключа

Асиметричні схеми. Такі ЕЦП відносяться до криптосистемам з відкритим ключем. На відміну від асиметричних алгоритмів шифрування, в яких шифрування проводиться за допомогою відкритого ключа, а розшифровка - за допомогою закритого, в схемах цифрового підпису підписання проводиться із

застосуванням закритого ключа, а перевірка підписи - із застосуванням відкритого ключа.

Рисунок 1.3 – Алгоритм роботи асиметричного ЕЦП

Недоліки асиметричної схеми ЕЦП маэ один недолік – продуктивність криптографічної схеми. Продуктивність асиметричної криптографічної схеми може виявитися недостатньою для відповідання поставленим вимогам. Рішенням для усунення даного недоліку є застосування спеціальної ефективно обчислюваної функції - функцією хешування чи хеш-функції.

На вхід функції подается повідомлення, а виході отримуется слово фіксованої довжини, яка буде набагато менше, ніж початкове повідомлення. Алгоритм ЕЦП не змінюється, але використовуется не саме повідомлення, а значення хеш-функції від нього. Це прискорює процес підпису та перевірки ЕЦП

1.2 Аналіз існуючих систем

Не дивлячись на велику кількість існучих алгоритмів електронно-цифровог підпису, программних реалізацій, які можна використовувати в інформаційно-телекомунікаційних системах не дуже багато. Адже більшість программ реалізації елетронно-цифрового підпису розроблені для використання для однієї конкретної компанії чи системи, або для використання в системах документообігу.

Для аналізу було обрано декілько відповідних систем:

1. КРИПТО-ПРО
2. КриптоАРМ
3. CryptoLibV2

КРИПТО-ПРО

КРИПТО-ПРО – засіб криптографічного захисту даних, який використовуються для виконання операцій шифрування/розшифровки та для реалізації методу електронно-цифрового підпису.

"КріптоПро CSP" призначений для:

- забезпечення конфіденційності даних та контролю їх цілісності за допомогою шифрування та імітозахисту
- забезпечення автентичності, конфіденційності та імітозахисту з'єднань по протоколу TLS
- контролю цілісності системного і прикладного програмного забезпечення для його захисту від несанкціонованих змін в коді програмного забезпечення і порушень правильності функціонування;
- управління ключовими елементами системи у відповідності з регламентом засобів захисту даних.

КРИПТО-ПРО за допомогою застосування удосконаленого алгоритму електронно-цифрового підпису дозволяє вирішити дві основні труднощі, які властиві "класичної" ЕЦП:

- відсутності докази моменту підпису;

- труднощі доведення статусу сертифіката відкритого ключа підпису на момент підпису (сертифікат може бути або дійсний, або анульований, або припинений).

Формат вдосконаленою підпису заснований на європейському стандарті CAdES.

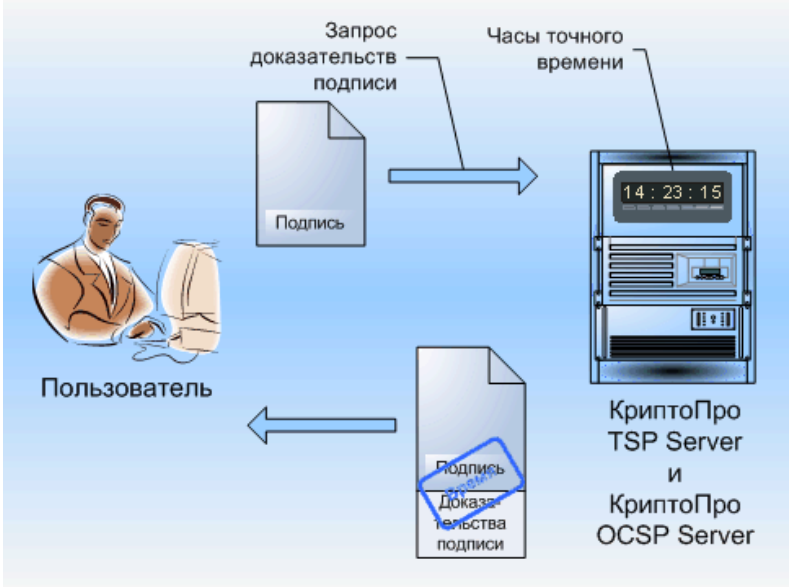

Рисунок 1.4 – Схема роботи КРИПТО-ПРО

Для доказу моменту підпису КРИПТО-ПРО використовує штампи часу, які відповідають міжнародної рекомендації RFC 3161 "Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)".

Докази дійсності сертифіката в момент підпису забезпечуються вкладенням в реквізити документа ланцюжка сертифікатів до довіреної УЦ і OCSP-відповідей. До цих доказів також додається штамп часу, що підтверджує їх цілісність в момент перевірки.

Великою перевагою КриптоПро є можливість перевірки достовірності ЕЦП без мережевих звернень. Це дозволяє не залежить від доступу до мережі, і працювати в оффлайн режимі.

КриптоАРМ

КриптоАРМ - це універсальна програма для шифрування і реалізації

електронно-цифрового підпису. Програма призначена для захисту корпоративної та особистої інформації, переданих по мережі Інтернет чи електронною поштою, збережених на знімних носіях (наприклад, на дисках, флеш-картах чи інших носіях).

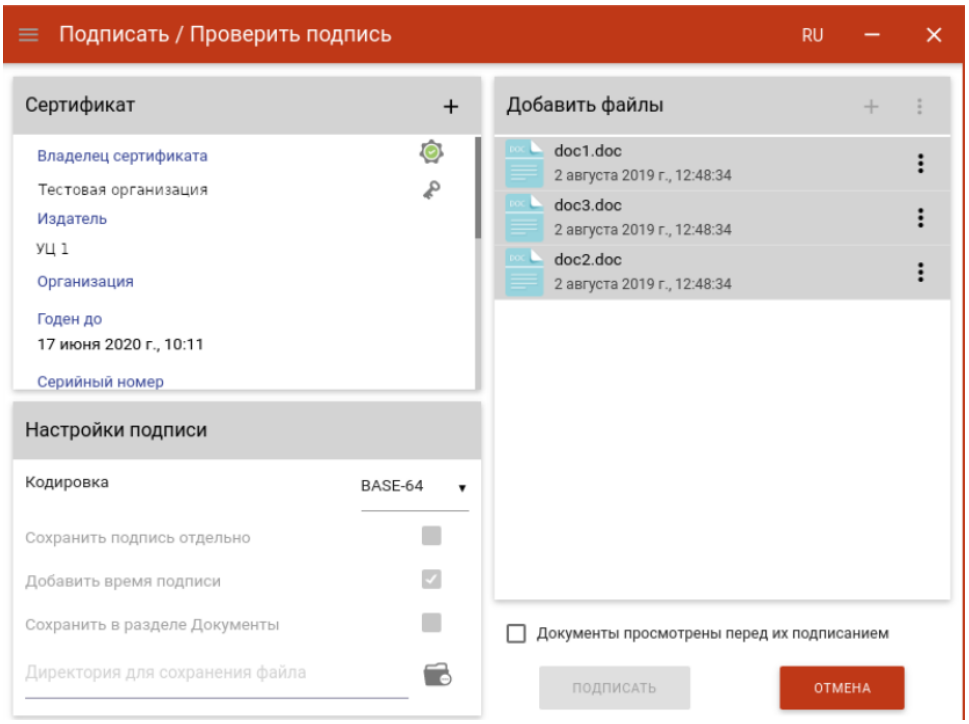

Рисунок 1.5 – Інтерфейс програми КриптоАРМ

КриптоАРМ використовується в інформаційних системах, де потрібно:

- надійно захистити даних від стороннього доступу;
- гарантувати цілісність даних при відправці по незахищених каналах зв'язку;
- забезпечити підтвердження справжності даних і авторства електронних даних;
- підтвердження часу підпису даних за допомогою штампа часу

КриптоАРМ має дуже зручний інтерфейс користувача, а також великий арсенал необхідних для інформаційних систем функцій.

КриптоАРМ підходить для інформаційних систем з великим потоком даних, так як він має можливість:

- Підписування даних як по одному, так і кілька великої кількості даних одночасно
- Підписування даних двома способами (окремо від вихідних даних і сумісно з даними)
- Розмір даних (які будуть підписані), обмежений тільки файлової системою
- Одночасна обробка необмеженої кількості файлів

Формат вдосконаленою підпису заснований на європейському стандарті CAdES Long.

Засіб електронного цифрового підпису «CryptoLibV2»

Засіб ЕЦП CryptoLibV2 призначене для криптографічного захисту даних з використанням електронного цифрового підпису (ЕЦП).

CryptoLibV2 являє собою набір бібліотек і допоміжного програмного забезпечення призначеного для шифрування і / або формування і перевірки ЕЦП.

Отриманий ЕЦП додається до первинних даних і забезпечує їх цілісність. Крім цього ЕЦП дозволяє ідентифікувати користувача, який підписав ці дані.

CryptoLibV2 складається з:

- бібліотеки взаємодії Avtor Cryptographic Provider.
- допоміжне програмне забезпечення.
- високорівневої бібліотеки інтеграції.

Завдяки бібліотеці Avtor Cryptographic Provider доступні до виконують такі криптографічні операції:

- генерацію особистих ключових даних відповідно до використовуючи алгоритм RSA
- імпорт / експорт ключових даних
- формування і перевірку ЕЦП на блок даних довільної довжини
- обчислення хеш-функції

- шифрування / розшифрування повідомлень використовуючи алгоритм RSA
- шифрування / розшифрування даних використовуючи Міжнародний алгоритм TDES і AES Програмно-технічний комплекс Центр сертифікації ключів «CryptoKDC»

ПТК ЦСК складається з:

- серверного та клієнтського програмно-технічного забезпечення
- програмного забезпечення інтеграції ЦСК в автоматизовані системи замовника;
- WEB портал надання допоміжних послуг клієнтам ЦСК

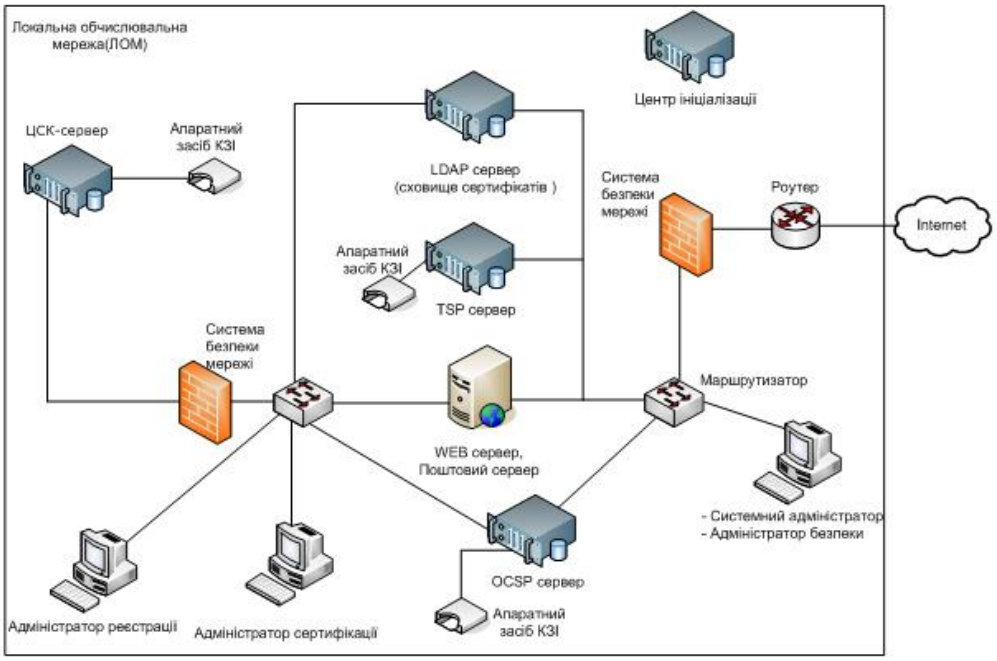

Рисунок 1.6 - Загальна схема підключення ПТК ЦСК до інформаційно-телекомунікаційної мережі

Програмно-технічний комплекс (ПТК) ЦСК забезпечує функціонування єдиної системи специфікацій на засоби криптографічного захисту інформації (КЗІ) і електронного цифрового підпису (ЕЦП) і надає організаційно-технічні засоби вживання сертифікатів відкритих ключів для вирішення наступних завдань інформаційної безпеки:

- створення системи управління ключами ЕЦП в системах автоматизації діяльності;
- побудови системи юридично значимого ЕЦП в системах електронного документообігу;
- забезпечення розподілу повноважень і надання доступу користувачам до інформаційних ресурсів автоматизованих інформаційних систем;
- контролю цілісності електронних документів, які передаються в автоматизованих системах замовника;
- забезпечення конфіденційності даних, які передаються в процесі інформаційного обміну.

ПТК ЦСК надає послуги сертифікації відкритих ключів шифрування і ЕЦП в корпоративних інформаційних системах.

ЦСК забезпечує можливість формування і обслуговування сертифікатів відкритих ключів шифрування і ЕЦП, які відповідають різним стандартам, а також використовувати алгоритм RSA/DSA з довжиною відкритого ключа 512-4096 біт.

Організаційна структура Центру сертифікації ключів забезпечує розподіл виконання адміністративних, технічних і технологічних операцій.

Висновок до розділу

У першому розділі дипломної роботи було проведено аналіз технології електронно-цифрового підпису для аутентифікації даних.

Окрім аутентифікації, електронно-цифровий підпис також може здійснювати контроль цілісності даних та захист даних від змін. Тобто електронно-цифровий підпис можна використовувати не тільки для аутентифікації, а й для підтвердження цілісності даних.

В результаті вивчення технології електронно-цифрового підпису було розглянуто спосіб її реалізації за допомогою симетричної та асиметричного підпису.

Симетричний спосіб має достатньо високу стійкість. Проте цей спосіб має дуже суттєвий недолік – підпис можна використовувати тільки один раз, бо після використання розкривається половина секретного ключа.

Тому було прийнято рішення обрати за основу асиметричну криптографічну схему, який дозволить реалізувати надійну аутентифікацію даних.

Також було розглянуто реалізації електронно-цифрового підпису, які можна використовувати в інформаційно-телекомунікаційній системі для аутентифікації даних.

2 РОЗРОБКА МОДЕЛІ

В даному розділі наведений опис розробки моделі, а саме приведені вимоги до підсистеми та теоретичний опис криптографічного алгоритму, який буде використовуватися в підсистемі.

2.1 Формування вимог до системи

Основною цілю підсистеми, яка буде розроблена, є аутентифікація данних користувачів, за допомогою ЕЦП з підвищенною криптостійкісттю, в автоматизованій інформаційно-комуннікаційній системі

До функціональних вимого можна віднести:

- Підсистема повинна мати змогу підписувати та перевіряти ЕЦП, як в автоматичному режимі, так і в режимі ручного керування
- Підсистема повинна проводити аналіз успішності аутентифікації даних, та вести записи у БД
- Підсистема повинна при непроходжені аутентифікації повідомляти про це оператору сервера.
- Підсистема повинна надавати змогу додавати нових користувачів до системи. Та у разі виникнення недовіри до користувача мати змогу заблокувати користувача

До нефункціональних вимого можна віднести:

- Підсистема повинна мати високий рівень криптостійкості
- Підсистем повина мати оптимальний рівень продуктивності
- Підсистем повинна вміти працювати з будь-якими текстовими данними
- Підсистем повинна бути проста в експлуатації
- Знання криптографічного алгоритма не повинно впливати на надійність процесу аутентифікації

2.2 Криптографічні схеми електронно-цифрового підпису

Електронно-цифровий підпис є одним з основних напрямків сучасної криптографії, відноситься до механізмів захисту цілісності даних.

Для кращого розуміння алгоритма роботи електронно-цифрового підпису було розглянуть декілька основних криптографічних схем.

Криптосхема Ель-Гамаля - це криптосхема з відкритим ключем, основанна на складності обчислення дискретних логарифмів в кінцевому полі. Саме с цієї схеми почався розвиток багатьох криптографічних схем електронно-цифрового підпису.

Генерація ключів:

1. Обираємо просте випадкове число p
2. Визначаємо випадкове мультиплікативний элемент α , при якому виконується умова $\alpha^{(p-1)} = 1 \bmod p$
3. Обираємо випадкове число x з інтервала (1, p) та взаємно просте з p
4. Обчислити $y = \alpha^x \bmod p$
5. В якості відкритого ключа використовується трійка чисел (y, α, p). Число x – секретний ключ, котрий необхідно зберігати в таємниці.

Генерація підпису (алгоритм підписання повідомлення M):

1. Обчислюємо дайджест повідомлення M: $m = h(M)$
2. Обираємо випадкове число $1 < k < p - 1$ взаємно просте з $p - 1$ та обчислюємо $r = g^k \bmod p$
3. За допомогою розширеного алгоритма Евклида обчислюємо число s, задовілняюче порівняння: $m \equiv (xr + ks) \bmod (p - 1)$
4. Підписом повідомлення M є пара (r, s)

Знаючи відкритий ключ (p, g, y), підпис (r, s) повідомлення M можно перевірити підпис.

Перевірка підпису:

1. Перевіряємо виконання умови: $0 < r < p$ та $0 < s < p - 1$. Якщо хоча б одне з цих умов не буле виконуватися, то підпис рахується не вірним.
2. Обчислення дайджеста: $m = h(M)$
3. Підпис рахується вірним, якщо виконується порівняння двох виразів: $y^r r^s \equiv g^m \bmod p$

Для того, щоб можно було б обчислити значення S, необхідно, щоб k було взаємно простим з числом $p - 1$. Ця вимога випливає з теореми про існування зворотнього єлемента. Саме тому при генерації параметра k повина виконуватися вимога:

$$НОД(k, p - 1) = 1$$

Криптосхема RSA. В основі криптографічної схеми з відкритим ключем RSA покладена задача множення та розкладання чисел на прості множники.

Генерація RSA-ключів:

1. Обчислюємо два випадкових числа p та q заданого розміру (наприклад, 1024 біти кожне)
2. Обчислюємо $n = pq$
3. Обчислюємо значення чисел функції Єйлера від числа n: $\varphi(n) = (p - 1)(q - 1)$
4. Обираємо ціле число e $(1 < e < \varphi(n))$, взаємно просте зі значенням функції $\varphi(n)$. За звичай в якості e беруть просте число, що містить невелику кількість одиничних бітів в двійковому записі, наприклад прості числа Ферма 17,257 або 65537. Число e називається відкритой єкспонентой. Час, необхідний для шифрування з використанням швидкого возведення в степін, пропорційний числу единичного біту в e. Зовсім мале значення e, наприклад 3, потенційно можуть ослабити безпеку схеми RSA.
5. Обчислюємо число d, мультиплікативно зворотньє к числу e по модулю $\varphi(n)$, тобто це число задовільняє умови:

$de \equiv 1 \bmod (\varphi(n))$, або $de \equiv 1 + k\varphi(n)$, де $k -$ деяке ціле число

Число d називається секретною експонентою. Зазвичай воно обчислюється за допомогою розширеного алгоритму Евклида.

Пара $P = (e, n)$ публікується в якості відкритого ключа RSA

Пара $S = (d, n)$ виконує роль секретного ключа RSA та тримається в секреті

Генерація підпису:

1. Беремо відкрите повідомлення M`
2. Створюємо електронно-цифровий підпис σ за допомогою секретного ключа (d, n): $\sigma = S_A(M`) \bmod n$
3. Передаємо пару $(M`, \sigma)$, яка складається з повідомлення та секретного ключа

Перевірка підпису:

1. Приймаємо пару $(M`, \sigma)$
2. Беремо відкритий ключ (e, n) сторони A
3. Перевіряємо справжність підпису:

$$P_A(\sigma) = \sigma^e \bmod n \equiv M` \rightarrow \text{підпис вірний}$$

При цьому повідомлення M` передається в не зашифрованому вигляді. Воно пересилається в початковому вигляді та його вміст не захищений.

Схема електронно-цифрового підпису на основі алгоритму Шнорра. Після публікації оригінальної роботи Єль-Гамаля зявилось декілька варіацій його алгоритму. Найбільш значущою серед них була схема електронно-цифрового підпису Шнорра. Ця схема являється варіантом схеми електронно-цифрового підпису Єль-Гамаля. Однако вона володіє властивістю, роблячи її цінною для криптографії з відкритим ключем: поле простих єлементів в цій схемі представлене набагато компактніше, ніж у схемі Єль-Гамаля. Проте, це призводить до зниження складності вирішення задачі на основі якої побудована дана криптографічна схема.

Більш компактною уявлення досягається за допомогою конструювання поля F_P, що містить набагато меншу підгрупу простого порядку q. В криптосистемі Ель-Гамаля безпечною довжиною параметра p вважається 1024. Більш того, з урахуванням розвитку технічних засобів, що дозволяють

вирішувати завдання дискретного логарифмування, ця величина буде поступово збільшуватися. Поява роботи Шнорра дозволило встановити величину | q | ~ 160. Вважається, що це величина не залежить від зростання p, оскільки інформація про підгрупі не відіграє значну роль при обчисленні дискретного логарифма в поле F_P.

Генерація системних параметрів:

1. Обираємо прості числа p, q, такі, щоб $|p| \approx 1024; |q| \approx 160$
2. Обрати єлементи $\alpha \in Z_p$ порядка q, тобто $\alpha^q = 1 \bmod p$
3. Обираємо криптографічну функцію хешування H
4. Генеруємо випадкове число k, таке, щоб $1 < k < q$
5. Обчислюємо значення $y = \alpha^{-x} \bmod p$
6. Закритим ключом являється x, а відкритий ключ зберігається в кортежі (p, q, α, y)

Генерація підпису для повідомлення M:

1. Генеруємо випадкове число k, таке, щоб $1 < k < q$
2. Обчислюємо значення $R = \alpha^k \bmod p$
3. До повідомлення M приєднуємо число R (M||R) та обчислюємо хеш-функціію H від отриманого значення:

$$E = H\,(M||R)$$

4. Обчислення параметру S:

$$S = k + xE \pmod q$$

5. Підпис к повідомленню M являється пара (R, S)

Перевірка підпису:

1. Обчислюємо значення R`:

$$R` = \alpha^S y^E \bmod p$$

2. До повідомлення приєднуємо число R` (M||R) та обчислюємо хеш-функцію H ві отриманого значення:

$$E` = H\,(M||R)$$

3. Порівнюємо два значення E` та E. Якщо воні однакові, то підпис справжній

Як і в схемі Єль-Гамаля параметр k, виконучий роль одноразового секретного ключа повинен обиратися за допомогою простого випадкового вибору та використовуватися лише один раз.

2.3 Спосіб побудови некоммутативних кінцевих груп векторів

Кінцеві кільца векторів можуть бути легко задані над кінцевим векторним простором шляхом визнгачення над веткорами деякої ассоціативної операції множення, використовуючи стандартну операцію додавання векторів, як додавання одноіменних координат. При цьому для заданого векторного простору є достатньо велика кількість різних варіантів визначення операції множення векторів, які задають кінцеві кільця з різними властивостями. Вектори розмірності m представлені у вигляді $(a, b, c, \ldots, q)$ або $a\boldsymbol{e} + b\boldsymbol{i} + \cdots + q\mathbf{w}$, де $\boldsymbol{a} + \boldsymbol{i} + \cdots . \mathbf{w}$ – деякі формальні базистні вектора; $a, b, c, \ldots, q$ – координати вектора, які є єлементами кінцевого поля $GF(p^s)$, де p – просте число, а s – степінь розширеного поля ($s \geq 1$).

Додавання векторів визначається по стандартній формулі:

$$(a, b, \ldots, q) + (x, y, \ldots, z) = (a + x, b + \mathrm{y}, \ldots, q + z)$$

Вектори вида εv, де $seGF(p^s)$ та $\mathrm{v} \in \{\mathrm{e}, \mathrm{i}, \ldots, \mathrm{w}\}$ – деякий формальний базисний вектор, представляють компоненти вектора.

Операція множення векторів визначається як попарне перемноження всіх компонентів пар-векторів по формулі:

$(a\boldsymbol{e} + b\boldsymbol{i} + \cdots + q\mathbf{w})^{\circ}(x\boldsymbol{e} + y\boldsymbol{i} + \cdots + z\mathbf{w}) \equiv axe^{\circ}e + aye^{\circ}i + \cdots + aze^{\circ}\mathbf{w} +$
$\mathrm{bx}\mathbf{i}^{\circ}e + byi^{\circ}i + \cdots + bzi^{\circ}w + \cdots qxw^{\circ}e + qyw^{\circ}i + \cdots + qzw^{\circ}w$

В даній формулі похідна пар базисних векторів замінюється на однокомпонентний вектор, наприклад εv, який заданий деякою таблицею множення базисних векторів. Координати однокомпонентних векторів, присутніх у таблиці множення базисних векторів, називаються структурними

коефіцієнтами. Якщо представлені за допомогою таблиці множення базисних векторів асоціативна операція множення є комутативною (некомутативною), то кінцевий векторний простір є векторним кінцевим комутативним (некомутативним) кільцем.

Побудова кінцевих кілець векторів є частиною в синтезу таблиці множення базисних векторів, яка визначає асоціативне множення. Питання синтезу комутативний ККВ досить добре вивчена. При парних розмірностях векторів m> 4 може бути застосований наступний загальний спосіб побудови таблиці множення базисних векторів, які задають некомутативними кінцевими кільцями векторів.

У першому рядку зліва направо записуються базисні вектора в деякій вихідній послідовності. У першому стовпчику зверху вниз записується та ж послідовність базисних векторів. У кожному парному рядку базисні вектора записуються, починаючи з елемента, зазначеного в першому стовпці, в зворотній послідовності. В непарних рядках базисні вектора записуються в тому ж порядку, як записані в першому рядку, тобто непарні рядки, починаючи з третього рядка, виходять шляхом циклічного зсуву першого рядка таблиці множення базисних векторів.

При m = 4 це побудова призводить до комутативної операції множення, а некомутативними забезпечується введенням структурних коефіцієнтів, кожен з яких дорівнює -1. При цьому структурні коефіцієнти розподіляються несиметрично відносно головної діагоналі таблиці множення базисних векторів, що проходить від лівого верхнього кута до правого нижнього кута. Приклад задання некомутативної таблиці множення базисних векторів наведена у таблиці 2.1.

Таблиця 2.1 - Побудова некомутативної ТМБВ для m = 4

°	**e**	**i**	**j**	**k**
e	**e**	**i**	**j**	**k**

i	**i**	p**ε**e	p**k**	**ε**j
j	**j**	**ε**k	**ε**p**e**	**ε**i
k	**k**	**ε**j	**ε**i	p**ε**e

Побудована таблиці множення базисних векторів для випадків m> 6 задає асоціативну і некомутативну операцію множення векторів. У випадках m>6 можуть бути знайдені симетричні і несиметричні розподіли структурних коефіцієнтів, які зберігають властивість асоціативності множення, проте не комутативність множення в першу чергу пов'язана з несиметричністю вихідної таблиці множення базисних векторів, тобто з вихідним несиметричним розподілом базисних векторів.

У таблиці 2.2 наведено приклад реалізації описаного вище загального способу побудови таблиці множення базиного вектора для випадку m = 8.

Таблиця 2.2 - Загальний спосіб побудови некомутативних ТМБВ для векторів з парним значенням розмірності

°	**e**	**i**	**j**	**k**	**u**	**v**	**w**	**x**
e	e	i	j	k	u	v	w	x
i	i	e	x	w	v	u	x	x
j	j	k	u	v	w	x	e	i
k	k	j	i	e	x	w	v	u
u	u	v	w	x	e	i	j	k
v	v	u	k	j	i	e	x	w
w	w	x	e	i	j	k	u	v
x	x	w	v	u	k	j	i	e

Загальним способом внесення структурних коефіцієнтів, кожен з яких дорівнює значенню $A \in GF(p^S)$, є його запис в кожну клітину, яка стоїть на перетині парного рядка і парного стовпчика. Однак описаний спосіб побудови

некомутативних кінцевих кілець веторів великої розмірності не забезпечує можливості отримання досить великих значень простого порядку векторів (порядком називається мінімальна натуральна ступінь, в яку потрібно звести зворотній вектор, щоб отримати одиничний вектор).

2.3 Синтез ТМБВ шляхом внесення несиметричного розподілу структурних коефіцієнтів

У випадку некомутативних груп чотиривимірних векторів операція множення задається за допомогою тамблиці множення базисних векторів, в якій базові вектора розподілені симетрично, а несиметричність, необхідна умовою некомутативності множення, пов'язана з несиметричним розподілом структурного коефіцієнта, рівного -1.

Дану таблицю множення базисних векторів можна розглянути як таблицю, отриману з деякого симетричного розподілу базисних векторів, на яке було накладено несиметричний розподіл структурного коефіцієнта. Останнє задало властивість некомутативності операції множення при збереженні якості асоціативності.

Це демонструє інший можливий підхід до синтезу некомутативних таблиць множення базисних векторів для випадку груп векторів великої розмірності. У такий способі синтез полягає в переборі великої кількості різних несиметричних розподілів структурного коефіцієнта і виборі таких розподілів, при яких зберігається властивість асоціативності множення, що були у вихідній таблиці множення базисних векторів, до внесення структурних коефіцієнтів.

Нехай задана деяка асоціативна вихідна таблиця множення базисних векторів довільної розмірності і кожна клітинка цієї таблиці може знаходиться в одному з двох станів умовних станів.

Умовно назвемо ці стану «Зайнято» і «Вільно». Всі клітинки вихідної таблиці знаходяться в стані «Вільно».

Випадковим чином вибирається осередок таблиці, в яку вноситься розтягуючий коефіцієнт (в якості розтягуючого коефіцієнта використовується «-1», так як таким чином може бути досягнута некомутативність операції асоціативного множення базисних елементів). Стан осередку таблиці змінюється на «Зайнято». Таким чином таблиця виводиться зі стану «рівноваги». Інша частина методу має рекурсивний характер. Базою рекурсії (елементарним завданням) буде асоціативна таблиця множення базисних векторів (рішення задачі), а так само неассоціативна таблиця множення базисних векторів, в якій всі осередки таблиці знаходяться в стані «Зайнято» (термінальний стан).

Крок рекурсії (спосіб зведення задачі до більш простих) полягає в тому, що вибирається трійка базисних елементів, для яких порушується асоціативність множення. З проміжних і кінцевих осередків таблиці, одержуваних при перемножуванні даної трійки обираються ті, які знаходяться в стані «Вільно» і, отже, в них можна внести структурний коефіцієнт. Стан даних осередків змінюється на протилежний і кожна можлива комбінація розстановки структурного коефіцієнта в обраних осередках таблиці, результатом якої буде відновлення асоціативності множення обраної трійки базисних векторів, породжує рекурсивний виклик.

Для досліждення роботи методу було обрано таблицю множення базисних векторів розміром 4x4. В таблиці 2.3 наведена вихідна комутативна таблиця множення базисних векторів розмірності 4.

Таблиця 2.3 - Вихідна коммутативна ТМБВ

°	**e**	**i**	**j**	**k**
e	e	i	j	k
i	i	e	x	w
j	j	k	u	v
k	k	j	i	e

Стан осередків таблиці буде вказуватися у окремій таблиці («С» - вільна, «З» - Зайнята). Спочатку все осередки знаходяться у вільному стані.

Таблиця 2.4 – Вихідна таблиця стану осередків таблиці

°	**e**	**i**	**j**	**k**
e	C	C	C	C
i	C	C	C	C
j	C	C	C	C
k	C	C	C	C

В довільний осередок таблиці множення базисного вектора вноситься розтягуючий коєфіцієнт, наприклад в осередок (i,i).

Таблиця 2.5 – ТМБВ з внесенням розтягуючого коефіцієнта

°	**e**	**i**	**j**	**k**
e	e	i	j	k
i	i	-e	x	w
j	j	k	u	v
k	k	j	i	e

Тепер таблиця множення базисного вектора не є асоціативною. Візьмемо одну з трійки базисних векторів, для яких порушується асоціативність: i, i, k. Змінимо стан проміжних осередків таблиці, що виникають під час множення базисних елементів, на стан «Зайнято» відповідно до схеми на рисунку 2.1.

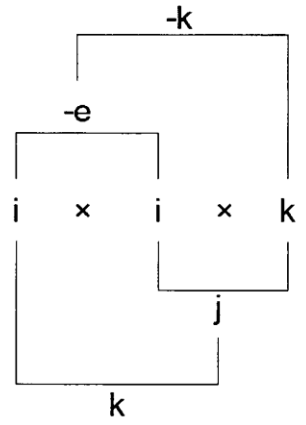

Рисунок 2.1 – Схема неасоціативного множення після першого кроку алгоритму

В результаті виконання першого шагу алгоритму отримуємо нову таблицю станів осередків таблиці.

Таблиця 2.6 – Таблиця станів осередків таблиці після першого кроку алгоритму

°	**e**	**i**	**j**	**k**
e	С	С	С	С
i	С	З	З	З
j	С	С	С	С
k	С	С	С	С

Відновити асоціативність вибраного базисного вектора можно за допомогою декількох варіатів:

1. Внести розтягуючий коефіцієнт в клітинку на перетині базисних векторів e та k
2. Внести розтягуючий коефіцієнт в клітинку на перетині базисних векторів i та k
3. Внести розтягуючий коефіцієнт в клітинку на перетині базисних векторів i та j

4. Внести розтягуючий коефіцієнт одночасно в клітинку на перетині базисних векторів e та k, i та k, i та j.

Проте, варіант 1 та 4 можна не розглядати, тому що експериментальні дослідження показали, що при внесенні растягуючого коефіцієнта в клітинки, що належать стовбцю або рядку при одиничному елементі, некомутативні векторні простори не утворюється.

Повторимо крок рекурсії для кожної з отриманих комбінацій розподілу структурного коефіцієнта. Як буде виглядати, таблиця множення базисних векторів, після застосування другого варіанта розподілу структурного коефіцієнта можно побачити в таблиці 2.7

Таблиця 2.7 – ТМБВ на другому кроці з внесенними розтягуючими коєфіцієнтами

°	**e**	**i**	**j**	**k**
e	e	i	j	k
i	i	-e	k	-j
j	j	k	e	i
k	k	j	i	e

Після виконання другого шагу алгоритму таблиця множення базисних векторів все ще не є асоціативної. Тому, візьмемо одну з трійок базисних векторів, для яких порушується асоціативність: j, i, i та змінимо стан проміжних осередків таблиці, що виникають під час множення базисних елементів, на стан «Зайнято».

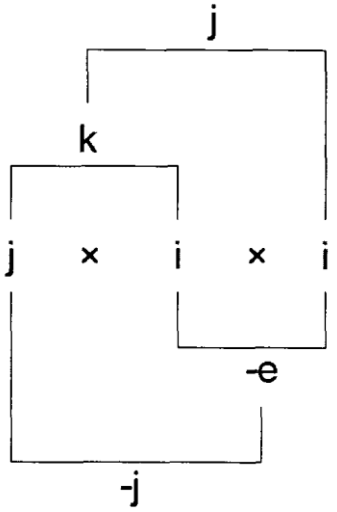

Рисунок 2.2 – Схема неасоціативного множення після другого кроку алгоритму

Відновити асоціативність обраних базисних векторів можна за допомогою наступних варіантів:

1. Внести розтягуючий коефіцієнт в клітинку на перетині базисних векторів j та i
2. Внести розтягуючий коефіцієнт в клітинку на перетині базисних векторів k та i

Після цього треба повторити шаг рекурсії для кожної з отриманих комбінацій розподілу структурного коєфіцієнта.

Таблиця 2.8 – Перший варіант ТМБВ після останнього кроку

°	**e**	**i**	**j**	**k**
e	e	i	j	k
i	i	-e	-k	j
j	j	k	-e	i
k	k	-j	i	-e

Таблиця 2.9 – Другий варіант ТМБВ після останнього кроку

°	**e**	**i**	**j**	**k**

e	e	i	j	k
i	i	-e	-k	j
j	j	k	e	i
k	k	-j	-i	e

Після отримання асоціативної таблиці множення базисних векторів, необхідно перевірити її на наявність коммутативності, так як операція множення, яка задає дану таблицю множення базисних векторів повинна бути некомутативною.

Даний метод був перевірений та використаний для випадку m = 8. Даний метод дозволяє знайти досить велике число (більше 600) різних несиметричних розподілів структурного коефіцієнта -1 для симетричної вихідної таблиці множення базисних векторів. Результат наведений у таблиці 2.10 та 2.11 відповідно.

Таблиця 2.10 – Перший приклад некоммутативних ТМБВ, отриманий методом несиметричного розподілу коєфіцієнтів

°	**e**	**i**	**j**	**k**	**u**	**v**	**w**	**x**
e	e	i	j	k	u	v	w	x
i	i	-e	k	-j	v	-u	x	-w
j	j	-k	e	-i	w	-x	u	-v
k	k	j	i	e	x	w	v	u
u	u	v	-w	-x	e	i	-j	-k
v	v	-u	-x	w	i	-e	-k	j
w	w	-x	-u	v	j	-k	-e	i
x	x	-w	-v	-u	k	j	-i	-e

Таблиця 2.10 – Другий приклад некоммутативних ТМБВ, отриманий методом несиметричного розподілу коєфіцієнтів

°	**e**	**i**	**j**	**k**	**u**	**v**	**w**	**x**
e	e	i	j	k	u	v	w	x
i	i	-e	k	-j	v	-u	x	-w
j	j	-k	e	-i	w	-x	u	-v
k	k	j	i	e	x	w	v	u
u	u	v	-w	-x	e	i	-j	-k
v	v	-u	-x	w	i	-e	-k	j
w	w	-x	-u	v	j	-k	-e	i
x	x	w	-v	-u	k	j	-i	-e

2.4 Алгоритм ЕЦП для кінцевих некомутативних груп векторів

Дослідження будови кінцевих некомутативними груп векторів розмірності $m = 4$ показало, що для побудови електронно-цифрового підпису слід скористатися підгрупою великого простого порядку q, що є дільником числа $p^2 - 1$.

Для того, щоб такий дільник знаходився у розкладі $p^2 - 1$, необхідно генерувати та використовувати прості числа p зі спеціальною структурою, а саме, таких що $p + 1 = 2q$, де q – просте число.

Генерація пари ключів:

1. Генеруємо просте число q розміром 160 біт.
2. Перевіряємо на простоту число $p = 2q - 1$. Якщо це число просте, то воно береться в якості модуля для задання вектора. Якщо воно не просте, то треба повернутися до шагу 1.
3. Беремо вектор G з розмірністю 4 і порядком $\omega(G) = q$
4. Обираємо випадкове число x. $|x| > 128$ біт
5. Визначаємо ветор $Y = G^x$. Число x є секретним ключем. Вектор Y – відкритим ключем

Підписання повідомлення:

1. Генеруємо випадкове число k
2. Визначаємо вектор $R = G^k$
3. Об’єднуємо підписане повідомлення M та визначений вектор R в одне повідомлення $M^* = M||R$
4. Визначаємо значення хеш-функції від M^*: $E = F_H(M^*)$. E – це перший елемент електронно-цифрового підпису
5. Обчислюємо другий елемент елетронно-цифрового підпису по формулі:

$$S = k + xE \bmod q$$

6. Підписом є пара чисел E та S

Перевірка підпису:

1. По відкритому ключу Y визначаемо вектор $R' = Y^{-E}G^S$
2. Обчислюэмо значення $e' = F_H = (M||R')$
3. Порывнюэмо значення e та e'. Якщо $e = e'$, то підпис для повідомлення M є справжнім. В протилежному випадку підпис не приймається.

2.5 Підвищення продуктивності криптосхему методом розпаралелювання операцій

Використання кінцевих груп векторів як в алгоритмах криптосхем дозволяє распараллеливать операції з ними за рахунок того, що кожна з координат результуючого вектора вважається окремо при виконанні операцій додавання і множення. Для приклада, взята процедура множення двох векторів розмірності 4, заданих над кільцем цілих чисел Zp^4 з використанням таблиці множення базового вектора:

$$(a\mathbf{e} + b\mathbf{i} + c\mathbf{j} + d\mathbf{k})°(u\mathbf{e} + x\mathbf{i} + y\mathbf{j} + z\mathbf{k})$$
$$= (au + bx + cy + dz)\mathbf{e} + (a\mathbf{x} + bu + cz + dy)\mathbf{i}$$
$$+ (ay + cu + dx + bz)\mathbf{j} + (az + du + by + cx)\mathbf{k}$$

Як можна бачити, обчислення кожної координати результуючого вектора

залежить тільки від вихідних даних. Таким чином, для отримання результату обчислення двох векторів необхідно обчислити 4 величини:

$$\begin{cases} au + bx + cy + dz \\ ax + bu + cz + dy \\ ay + cu + dx + bz \\ az + du + by + cx \end{cases}$$

Обчислення кожної координати потребує в даному випадку 4 множення чисел довжиною p біт по модулю p, який в загальному випадку дорівнює $p^{a/n}$, де n – розмірність вектора, а a - коефіцієнт розширення поля.

Операція додавання практично не впливає на час, що витрачається обчислення, тому нею можна знехтувати.

При збільшенні розмірності вектора, розмір його координат зменшується пропорційно, що дозволяє знижувати час на операції в базовому поле, так як через особливості архітектури комп'ютера, багато операцій з маленькими числами вигідніше за часом, ніж одна операція з великим числом.

Для перевірки ефективності методу розпаралелювання базових операцій було проведено долідження для визначення часу, що витрачається на множення двох векторів по модулю в двох випадках: без розпаралелювання операцій і з розпаралелюванням. Для приклада були взяті вектора розмірності 4, 6 і 8, сумарний розмір координат яких дорівнював 1024 біта, а також окремий випадок 16-мірного вектора розміру 512 біт. На малюнку 6 зображені діаграми порівняння часових витрат на 100000 операцій множення.

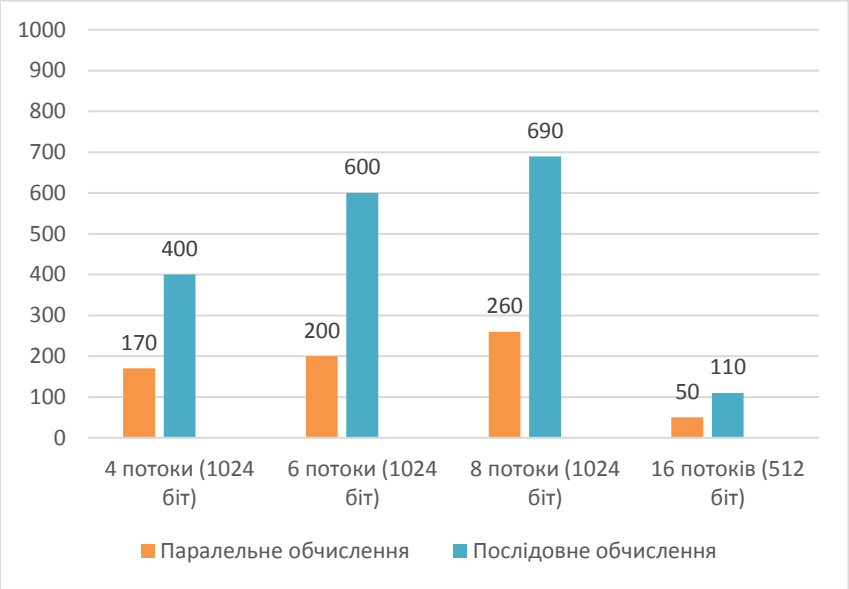

Рисунок 2.3 - Оцінка ефективності розпаралелювання операцій в кінцевих некомутативних групах векторів

Як видно з вищенаведених діаграм, розпаралелювання операцій дозволяє скоротити час, що витрачається на множення двох векторів в 2-2.5 рази. Окремо варто відзначити високу продуктивність у випадку з розмірністю вектора 16. Вона обумовлена тим, що розмір кожної координати 16-мірного вектора довжиною 512 біт дорівнює 32 бітам, що дозволяє швидко перемножать їх в регістрах процесора без використання додаткових алгоритмів, на відміну від випадку, коли розмір координат більше розміру регістрів.

Отримані експериментальні результати виявилися нижчими теоретичного значення, рівного n разів, де n - розмірність вектора. Це обумовлено додатковими тимчасовими витратами на розпаралелювання обчислень

В цілому, враховуючи особливості апаратної реалізації алгоритмів і рекомендації до вибору таблиці множення базового вектора для некомутативних кінцевих груп, можна домогтися зниження часу, що витрачається на обчислення в кінцевих некомутативних групах векторів приблизно в n / 2 раз, де n - розмірність векторів групи.

Висновок до розділу

В даному розділі було розглянуто алгоритми криптографічних схем, які на даний час широко використовуються при реалізалізації елетронно-цифрового підпису.

Була розглянута криптосхема Ель-Гамалля, криптосхема RSA та електронно-цифровий підпис на основі алгоритма Шнорра. Проте в умовах швидкого росту обчислювальних можливостей криптографічна стійкість даних схем зменшується. Тому для підтримки високого рівня стійкості необхідно створювати ключі великої довжини. Наприклад, на даний час при використанні криптографічної схеми RSA найменша довжина ключів, яка забезпечує достатню стійкість - 1024 біти.

Для зменшення довжини ключів зі збереженням рівня криптостійкості було розглянуто можливість використання некомутативних груп в реалізації електронно-цифрового підпису.

Було розглянуто загальні способи задання кінцевих груп векторів на прикладі некомутативних кінцевих векторів.

Було досліджено метод синтеза некомутативних таблиць множення базових векторів, які будуть задавати некоммутативні групи векторів. Для цього було використано метод несиметричної несимметричної ростановки коєфіцієнтів. Також в даному розділі представлен алгоритм електронно-цифрового підпису та спосіб підвищення продуктивності даного алгоритму за допогою розпаралелення процесу розрахунку.

3 ПРОЕКТУВАННЯ ПІДСИСТЕМИ

В даному розділі приведений опис розробленої підсистеми. Приведений опис структурної та функціональної схем, UML – діаграм.

3.1 Структурна схема підсистеми

Структурна схема підсистеми представлена в додатку А.

Система складається з шісти модулів:

- Модуль серверної частини аутентифікації
- Модуль інтерфейсу оператора
- Модуль інтерфейсу користувача
- Модуль ідентифікації
- Модуль аутентифікації
- Модуль електронно-цифрового підпису

Модуль серверної частини аутентифікації знаходиться на серверній частині підсистеми та відповідає за перевірку електронно-цифрового підпису. Також цей модуль взаємодіє з базою даних.

База даних призначена для зберігання даних необхідних для ідентифікації та аутентифікації користувачі, а також ведення журналу перевірки даних отриманих сервером від користувачів.

Модуль інтерфейса оператора використовується для взаємодії оператора з підсистумою, контролю за перебігом перевірки підписів.

Модуль інтерфейсу користувача необхідний для взаємодії користувача з підсистемою, Модуль інтерфейсу користувача спілкується з модулями, які знаходятся на клієнтській стороні на программному рівні. Також даний модуль спілкується з модулем серверної частини аутентифікації за допомогою спеціального захищеного каналу звязку, який створює система за допомогою засобів протоколів TCP/IP.

Модуль ідентифікації виконує ідентифікацію клієнта при першому

використанні підсистеми та за необхідністю.

Модуль аутентифікації знаходиться на клієнтській частині підсистеми та відповідає за підписання даних електронно-цифровим підписом.

Модуль електронно-цифрового підпису створює нові пари ключів для реалізації електронно-цифрового підпису, слідкує за їхнім сроком дійсності. При необхідності даний модуль оновлює пару ключів.

3.2 Розробка функціональної схеми

Функціональна схема підсистеми представлена в додатку Б.

Для більшої наглядності функціональних можливостей підсистеми на схемі були зображені компоненти основних модулів системи.

Модуль серверної частини аутентифікації складається з трьох компонентів:

1. Алгоритма перевірки електронно-цифрового підпису
2. Ідентифікації користувачів
3. Роботи з базою даних

Алгоритм перевірки електронно-цифрового підпису виконує безпосередньо функцію аутентифікації даних. Після перевірки підпису усі результати даної перевірки передаються в базу даних. При виявлені підробки даних, про це повідомляється оператору інформаціно-телекомунікаційної системи.

Ідентифікацію користувач проходить при першому використанні системи та при винекненні необхідності під час подальшої роботи. Функцією даного компонента є перевірка індентифікаторів та прав доступу до сервера користувачів. У разі успішного проходженням користувачем процедури ідентифікації дані про користувача вносяться до бази даних.

На клієнтській частині підсистеми основним та найбільш функціональним є модуль інтерфейсу користувача. За допомогою цього модулю користувач може користуватися усіма необхідними функціями.

Модуль інтерфейсу користувача складається з трьох компонентів:

1. Налаштування облікового запису
2. Підписання даних
3. Менеджер ключів

Через компонент налаштування облікового запису клієнт має доступ до модулю ідентифікації та може пройти ідентифікацію у підсистемі для подальшої роботи у ній.

Для аутентифікації своїх даних в інформаційно-телекомунікаційній системі користувач повинен їх підписати та надіслати на сервер для перевірки. Відправкою даних займається інформаційно-телекомунікаційна система, а от підписання данних – це функція модуля аутентифікації, який можна запустити через інтерфейс користувача.

Найбільш функціональним компонентом інфтерфейсу користувача є менеджер ключів. Менеджер ключів повність відповідає за коректну роботу ключів, за допомогою, яких працює електронно-цифровий підпис.

Після проходження ідентифікації, для початку роботи підсистеми аутентифікації клієнту необхідно створити пару ключів. Це робиться за допомогою менеджера ключів, який за допомогою модуля генерації електронно-цифрового підпису сворює цю пару ключів.

Також менеджер ключів за допомогою цього модуля слідкує за сроком дії ключа. У випадку, коли срок дії елетронно-цифрового підпису сплив чи виникла необхідність замінити ключі, менеджер ключів оновлює їх.

3.3 Розробка сценарія використання

Діаграма сценарія використання підсистеми аутентифікації даних представнена в додатку В.

В системы представлено два типа користувачів.

Перший тип – це користувач клієнтської частини підсистеми, який після ідентифікації в системі, може створювати ЕЦП та використовувати його для підпису своїх даних.

Другий тип – це оператор серверу, який проводить аутентифікацію даних користувача, зберігає результати в базі даних та реагує на дані, які не пройшли аутентифікацію.

Далі проводиться опис всіх прецидентів сценарія використання підсистеми у вигляді таблиць.

Опис сценарію використання підсистеми «Авторизація користувача в підсистемі» представлений в таблиці 3.1

Таблиця 3.1 – Опис сценарію використання підсистеми «Авторизація користувача в підсистемі»

Назва	Авторизація користувача в підсистемі
ID	1
Опис	Користувач при першому запуску підсистеми повинен пройти процедуру авторизації
Актори	Користувач
Частота використання	Низька
Тригери	Натискання на кнопку авторизації
Передумова	Користувач не авторизований; Наявність на екрані форми авторизації
Постумова	Користувач отримує доступ до підсистеми
Головний курс	1. Після первшого запуску підсистеми відображається форма авторизації 2. Користувач водить свій ID та пароль в відповідні поля 3. Користувач натискає кнопку авторизації

	4. Перевірка ведених даних 5. Користувач входить в систему
Альтернативний курс	1. Після первшого запуску підсистеми відображається форма авторизації 2. Користувач водить свій ID та пароль в відповідні поля 3. Користувач натискає кнопку авторизації 4. Дані введені не відповідають тим, які зберігаються в базі даних 5. Виводиться повідомлення про неправильно введені данні
Винятки	Користувач не правильно ввів данні чи взагалі не має доступу до підсистеми

Опис сценарію використання підсистеми «Управляння користувачами» представлений в таблиці 3.2

Таблиця 3.2 – Опис сценарію використання підсистеми «Управляння користувачами»

Назва	Управляння користувачами
ID	2
Опис	Оператор серверу може створювати та видаляти користувачів підсистеми
Актори	Оператор
Частота використання	Середня
Тригери	Натискання кнопки для управління користувачами підсистеми

Передумова	У сервера повинен бути користувач, який уже авторизований чи який пройшов процедуру авторизації в підсистемі
Постумова	Доданий чи заблокований користувач
Головний курс	1. Оператор серверу натискає кнопку додати дані про користувача 2. Оператор вводить необхідні дані 3. Программа перевіряє чи є такий користувач в підсистемі 4. Якщо такого користувача немає, то його дані записуються до бази даних. Після чого він є користувачем системи 5. Якщо такий користувач є і ведені дані відрізняються від тих, що зберігаються в базі даних, в оператора сервера запитується, чи дійсно він хоче перезаписати дані користувача
Альтернативний курс	1. Оператор серверу натискає кнопку заблокувати доступ користувача до системи 2. Виводиться список авторизованих користувачів 3. Оператор обирає користувача, якого він хоче видалити 4. Виводиться повідомлення про необхідність підтвердити свою дію 5. Після підтвердження дії, всі дані про користувача будут видалені з бази даних.

	Публічні ключі даного користувача також будуть видалені
Винятки	В підсистеми немає користувачів чи користувач

Опис сценарію використання підсистеми «Створення публічного та секретного ключа» представлений в таблиці 3.3

Таблиця 3.3 – Опис сценарію використання підсистеми «Створення публічного та секретного ключа»

Назва	Створення публічного та секретного ключа
ID	3
Опис	Користувач може згенерувати пару ключів для ЕЦП: публічний та секретний ключ
Актори	Користувач
Частота використання	Низька
Тригери	Натискання кнопки для створення ключів для реалізації ЕЦП
Передумова	Користувач повинен бути авторизованим в підсистемі
Постумова	Створена нова пара ключів для ЕЦП
Головний курс	6. Клієнт натискає кнопку для генерації ключів для реалізації ЕЦП: публічний та секретний ключ 7. Підсистема генерує ключі

	8. На екрані з'являється повідомлення про успішне створення ключів
Альтернативний курс	6. Клієнт натискає кнопку для генерації ключів для реалізації ЕЦП: публічний та секретний ключ 7. У даного клієнта вже створена пара ключів 8. На екрані з'являється повідомлення про наявність пари ключів та пропозицію згенерувати нову пару 9. Користувач обирає варіант подальших дій
Винятки	Користувач вже має пару ключів, і відмовляється від створення нових

Опис сценарію використання підсистеми «Зберігання публічних ключів користувачів» представлений в таблиці 3.4

Таблиця 3.4 – Опис сценарію використання підсистеми «Зберігання публічних ключів користувачів»

Назва	Зберігання публічних ключів користувачів
ID	4
Опис	Підсистема зберігає публічні ключі користувачів, які використовуються для перевірки підпису
Актори	Оператор
Частота використання	Середня
Тригери	Сигнал від системи про отримання публічних ключів від користувача

Передумова	Отримання ключів від користувача
Постумова	Зберігання публічних ключів користувачів на сервері
Головний курс	1. Отримуємо сигнал від системи, про отримання публічного ключа від сервера 2. Оновлюємо статус користувача в базі даних 3. Переміщуємо публічний ключ в місце, де він буде зберігатися
Альтернативний курс	-
Винятки	-

Опис сценарію використання підсистеми «Підписання файлів» представлений в таблиці 3.5

Таблиця 3.5 – Опис сценарію використання підсистеми «Підписання файлів»

Назва	Підписання файлів за допомогою секретно ключа
ID	5
Опис	Користувач може підписати свої файли за допомогою секретного ключа
Актори	Користувач
Частота використання	Висока
Тригери	Натискання кнопки підписання файлів; в автоматизованій системі викликається самою системою

Передумова	Користувач повинен мати пару ключів для реалізції електронно-цифрового підпису
Постумова	Документ підписаний
Головний курс	1. Користувач натискає кнопку для підписання ключів 2. Обирає файли, які необхідно підписати 3. Користувач отримує повідомлення про успішне підписання файлів
Альтернативний курс	1. Автоматизована система передає сигнал про необхідність підписати файли, які в подальшому будуть передані на сервер 2. Підсистема підписує файли 3. Файли готові до відправки на сервер
Винятки	Користувач вже має згенерованої пари ключів чи інсуючі ключі не дійсні

Опис сценарію використання підсистеми «Перевірка підпису» представлений в таблиці 3.6

Таблиця 3.6 – Опис сценарію використання підсистеми «Перевірка підпису»

Назва	Перевірка підпису
ID	6
Опис	Оператор сервера може перевірити справжність отриманих від користувача даних
Актори	Оператор
Частота використання	Висока
Тригери	Натискання кнопки для перевірки даних

	отриманих від користувача
Передумова	На сервері повині бути підписані користувачем дані
Постумова	Повідомлення про справжність отриманих даних
Головний курс	1. Оператор сервера натискає кнопку для перевірки даних отриманих від користувача 2. Оператор обирає, які файли він хоче перевірити 3. Програма перевіряє всі обрані файли 4. Запускається сценарій “Веденя журналу результатів аутентифікації” 5. Якщо файл по якійсь причині не пройшов перевірку то, про це повідомляється оператору сервера, який приймає рішення. На час прийнятя рішення файли не передаються системі, а поміщаються в карантин.
Альтернативний курс	-
Винятки	На сервері немає файлів користувача

Опис сценарію використання підсистеми «Веденя журналу результатів аутентифікації» представлений в таблиці 3.7

Таблиця 3.7 – Опис сценарію використання підсистеми «Веденя журналу результатів аутентифікації»

Назва	Веденя журналу результатів аутентифікаці
ID	7
Опис	В підсистемі ведеться журнал даних з результатами аутентифікації
Актори	Оператор
Частота використання	Висока
Тригери	Викликається підсистемою під час перевірки підпису
Передумова	Найявність підписаних файлів
Постумова	Запис результатів аутентифікації в базі даних
Головний курс	4. Запис результатів аутентифікації вбазу даних
Альтернативний курс	-
Винятки	-

Опис сценарію використання підсистеми «Контроль за сроком дії ключів» представлений в таблиці 3.8

Таблиця 3.8 – Опис сценарію використання підсистеми «Контроль за сроком дії ключів»

Назва	Контроль срока дії ключа
ID	8
Опис	Підсистема слідкує за сроком дії ключів і при закінчені терміну дії створює нові

Актори	Користувач
Частота використання	Висока
Тригери	Щоденне спрацювання, по розкладу
Передумова	Користувач повинен мати пару ключів для реалізції електронно-цифрового підпису
Постумова	Уникнення припинення роботи підсистеми у разі закінчення сроку дії пари ключів
Головний курс	5. По розкладу, підсистема запускає алгоритм перевірки срока дії пари ключів 6. Якщо ключі дійні, то алгоритм припиняє свою роботу
Альтернативний курс	1. По розкладу, підсистема запускає алгоритм перевірки срока дії пари ключів 2. Якщо срок дії ключів вишов, про це повідомляється користувачеві, якій приймає рішення, що далі робити
Винятки	-

3.4 Опис системи в якій працюватиме підсистема

При розробці програмного забезпечення для реалізації підсистеми аутентифікації в першу чергу необхідно визначити в якій системи вона буде працювати.

Дана підсистема буде працювати існуючій автоматизованій інформаційно-телекомунікаційній системі. Ця система для зв'язку між собою клієнтських комп'ютерів і серверів використовує просту модель взаємодії "клієнт-сервер". Тобто, клієнти - ініціалізують з'єднання, а сервера – очікують отримання

сигналу від своїх клієнтів.

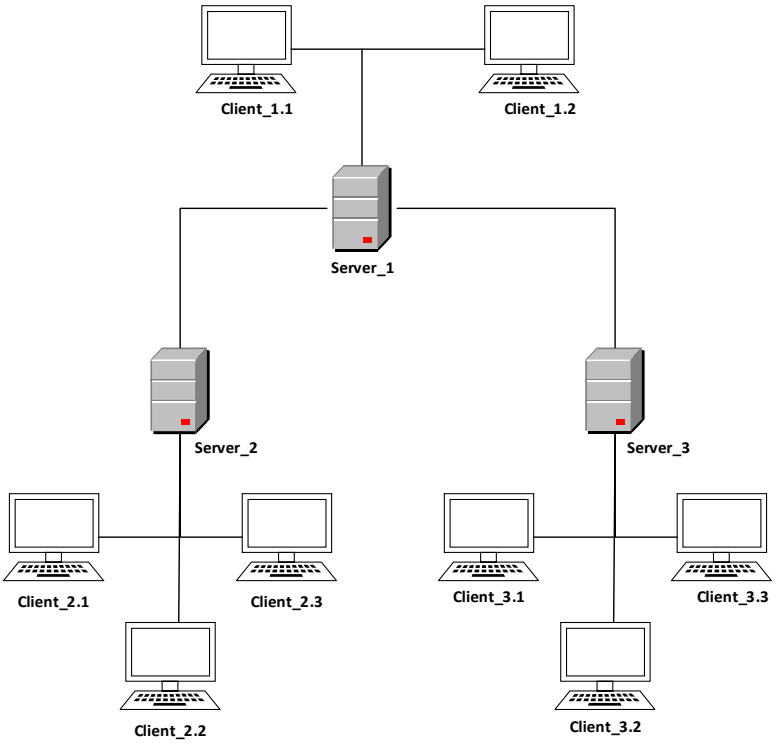

Рисунок 3.1 – Топологія інформаційно-телекомунікаційної системи

Інформаційно-телекомунікаційна система складається з центрального офісу та двох філіалів. Між серверами та клієнтами існує захищений канал зв’язку по якому автоматично передаються дані.

Дана система була розроблена для компанії, яка займається анкетуванням. Головний офіс передає нові підготовлені питання для анкетування на свій сервер. До сервера головного офісу підключається сервер одного з філіалів. Він забирає бланки з питаннями, та передає вже заповненні анкети, бланки який були передані раніше.

Анкетування проводиться за допомогою спеціальної програми. Яка задає питання, отримує відповідь, та передає ці дані назад на сервер, де проводиться аналіз анкет по спеціальному алгоритму.

Проте час від часу в даній системі бувають перебої в роботі по причинам невідповідності даних.

Причинами цього може бутите, що:

1. Користувач не використовував програму проходження анкетування, а вписав дані в анкету самостійно
2. Користувач після проходження тестування змінив значення в готовій анкеті. Це можливо у випадку, коли користувач працював в офлайн режимі, і дані одразу не передались на сервер, або ж з технічних причин не було зв’язку з сервером. У такому разі дані знаходяться у клієнта, до моменту коли він зможе підключитися до сервера, і вони автоматично будуть передані.

Для вирішення даної проблеми була розроблена підсистема аутентифікації даних користувачів.

3.5 Вибір та обґрунтування елементів та технологій

В якості мови програмування для підсистеми аутентифікації було обрано мову програмування Python.

Python - інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією. Python підтримує структурне, об'єктно-орієнтоване, функціональне програмування та має велику кількість стандартних бібліотек та багато спеціальних, додаткових модулів, які можна при необхідності легко підключити.

Першою, великою перевагою даного мови програмування є його простота. Навіть складний алгоритм реалізований за допомогою Python може мати стислий та лаконічний вихідний код, при цьому програміст затрачує менше часу та зусиль для досягнення поставленої цілі.

Одним з важливих критеріїв, чому був обраний саме Python - це те, що він універсальний, підтримує функції та методи роботи зі списками, кортежами, векторами.

За допомогою даної мови програмування можна легко розробити мультиплатформову систему, адже він працює з усіма операційними системами та файлової системи також. Для цього існує окремий модуль, Який

реалізує безліч функцій для роботи з операційною системою, причому їх поведінка, як правило, не залежить від ОС, тому програми легко переносяться.

Python може виконувати практично будь-яке завдання за рахунок легкого підключення та використання модулів. Також до основної програми, написаної на Python, при необхідності можна інтегрувати функції, які були реалізовані на C++. Використовуючи такі готові конструкції, можна вирішувати складні завдання, пов'язані з обробкою графіки, математичними обчисленнями, візуальним моделюванням.

Але крім переваг, ця мова програмування має і недоліки. Однією з них являєтеся продуктивність. Навіть в порівнянні з іншими мовами він може програвати. Однак на при справді, завдань, де важлива супер-швидкість реалізації коду, не так багато. Та й продуктивність можна підвищити за допомогою спеціальних модулів або за допомогою написання складних функцій на більш продуктивній мові програмування та інтеграції в основний код написаний на Python.

Також другим, невеликим недоліком є те, що програмісту, що який буде працювати з цією мовою, доведеться враховувати, що для гідної роботи необхідно створювати максимально логічний і простий код.

Numeric Python (NumPy) - це кілька модулів для обчислень в багатовимірних масивах, необхідних для багатьох чисельних додатків. Особливо підкреслимо відміну масиву від набору даних (списку чи кортежу). В даній бібліотеці підтримуються багатовимірні масиви (включаючи матриці) та високорівневі математичні функцій, призначені для роботи з багатовимірними масивами.

Математичні алгоритми, реалізовані на інтерпретованих мовах (наприклад, Python), часто працюють набагато повільніше тих же алгоритмів, реалізованих на компільованих мовах (наприклад, C, Java). Бібліотека NumPy надає реалізації обчислювальних алгоритмів (у вигляді функцій і операторів), оптимізовані для роботи з багатовимірними масивами.

Однією з найбільш дискутованих особливостей Python є GIL - Global

Interpreter Lock. GIL не дозволяє в одному інтерпретаторі Python ефективно використовувати більше одного потоку. Захисники GIL стверджують, що однопоточні програми при наявності GIL працюють набагато ефективніше. Але наявність GIL означає, що паралельні обчислення з використанням безлічі потоків і загальної пам'яті неможливі. А це досить сильне обмеження в сучасному багатоядерному світі.

Один із способів подолання GIL за допомогою потоків на C ++. Проте, щоб не писати окрему функцію на C ++ можно скористатися модулем multiprocessing.

Multiprocessing дозволяє працювати з процесам як з потоками. Це означає, що модуль бере на себе проблему синхронізації окремих Python-процесів.

Для візуалізації функціоналу підсистеми та легкого використання функцій користувачами необхідно розробити інтерфейс. В Python, це можна реалізувати за допомогою PyQt та Qt Designer.

PyQt - це набір «прив'язок» графічного фреймворка Qt для мови програмування Python, виконаний у вигляді розширення Python.

PyQt практично повністю реалізує можливості Qt. PyQt також включає в себе Qt Designer (Qt Creator) - дизайнер графічного інтерфейсу користувача. Програма pyuic генерує Python код з файлів, створених в Qt Designer. Це робить PyQt дуже корисним інструментом для швидкого прототипування. Крім того, можна додавати нові графічні елементи управління, написані на Python, в Qt Designer.

Останьє, що треба обрати для реалізації підсистеми аутентифікації – це база даних. Для даної підсистеми було обрано MySQL.

MySQL – це всім відома реалізація системи управління базами даних. Вона проста в установці, працює нормально без особливих налаштувань. При належному підході може гнучко налаштовуватися під будь-які потреби. Також MySQL може підтримувати роботу БД значних розмірів. MySQL працює на різних платформах та легко переноситься з однієї платформи на іншу. MySQL має систему контроля доступу до даних, забезпечує шифрування даних.

Висновок до розділу

У даному розділі було спроектовано підсистему аутентифікації даних.

Була розроблена структурна та функціональні схеми.

При розробці підсистеми було враховано те, що дана підсистема повина працювати в автоматизованій інформаційно-телекомунікаційній системі.

На структурній схемі зображені всі елементи підсистеми. Підсистема буде складатися з серверної та клієнтської частина. На сервері знаходиться база даних, серверна частина аутентифікації та інтерфейс оператора серверу. Сервер виконує функцію перевірки підписаних даних та контроль процеса аутентифікації цих даних.

На клієнті знаходиться інтерфейс користувача, за допомогою якого користувач спілкується з підсистемою. Користувач створює та управлює парою ключів, які використовуються для реалізації електронно-цифрового підпису.

Також був розроблений сценарій використання. За допомогою таблиць було описано поведінку підсистеми при взаємодії з нею користувача та оператора серверу.

Перед початком реалізації підсистеми обрано мову програмування та необхідні модулі та технології для реалізації спроектованої підсистеми. Підсистема аутентифікації даних буде реалізована за допомогою мови програмування – Python. Дана мова программування дозволить легко та швидко реалізувати всі модулі підсистеми. Python має необхідні модулі для математичних обрахунків, роботи з векторами і матрицями, а також можливість паралельного їх розрахунку.

4 РЕАЛІЗАЦІЯ ПІДСИСТЕМИ

Підсистема аутентифікації даних користувача була розроблена для автоматизованої інформаційно-телекомунікаційної системи, проте її можна використовувати, як в звичайних інформаційно-телекомунікаційних системах, так і для особистого використання.

Підсистема аутентифікації даних користувачів складається з двох частин:

1. Серверна частина – відповідає за перевірку даних
2. Клієнтська частина – відповідає за підписання даних (файлів)

В додатку Г зображена діаграма послідовності на якій відображається робота від початку авторизації користувача до перевірки підписаних даних.

4.1 Реалізація серверної частини підсистеми

На всіх серверах системи встановлюється серверне ПЗ, розроблене для реалізації підсистеми аутентифікації.

Підсистема при роботі в обраній інформаційно-телекомунікаційній системі не потребує установки та налаштувань. Для початку роботи підсистеми необхідно просто запустити відповідне ПЗ і программа самостійно все встановить. При роботі в іншій системі необхідно буде створити базу даних з усіма необхідними таблицями та зі списком користувачів системи та даними про них.

В автоматизованій інформаційно-телекомунікаційній системі всі таблиці створюються автоматично, дані про користувачів беруть із системи. Всі інші таблиці будуть заповнюватись під час роботи підсистеми.

Після налаштувань можно починати працювати з серверним ПЗ підсистеми через консоль чи через інтерфейс оператора серверу.

Для авторизації оператор сервера використовує свої дані зап допомогою яких він входить в інформаційно-телекомунікаційну систему.

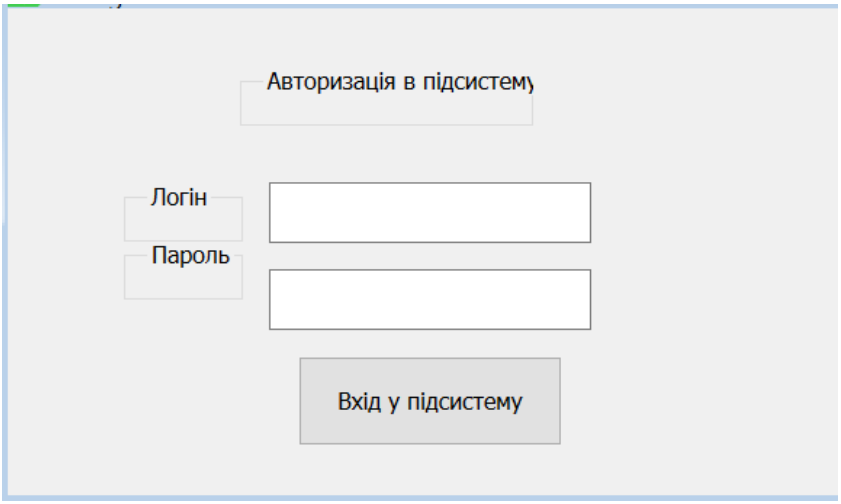

Рисунок 3.2 – Вікно авторизації оператора сервера

При встановлені серверного ПЗ в інформаційно-телекомунікаційній системі в базу даних серверу записуються дані про користувачів даного серверу і підсистема чекає, коли клієнти почнуть підключатися та проходити авторизацію в підсистемі.

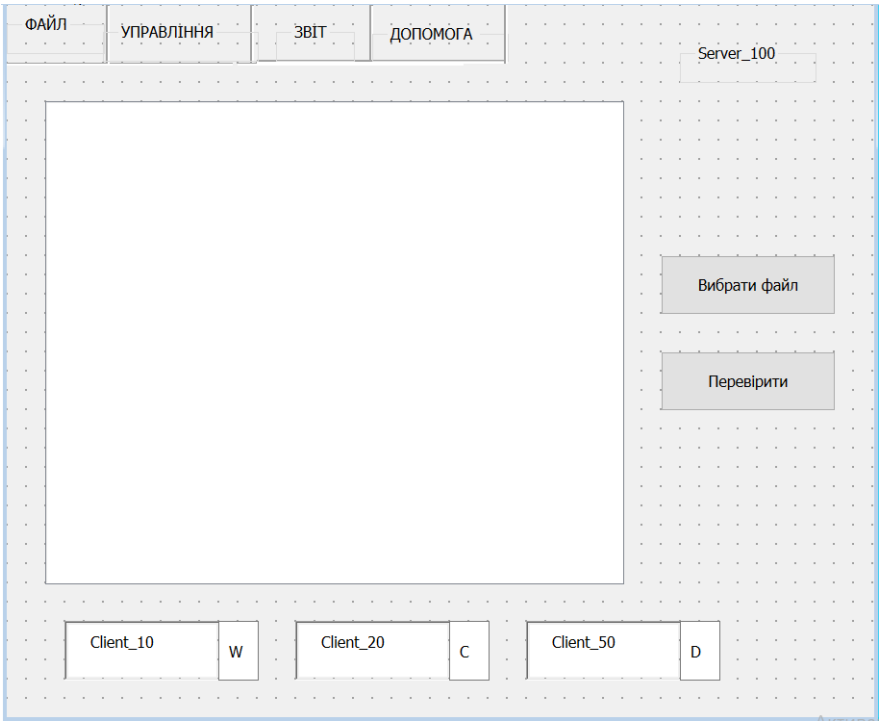

Рисунок 3.3 – Інтерфейс користувача сервера

Також оператор сервера має змогу додавати та видаляти користувачів

системи. Це робиться за допомогою: Верхне меню --> Управління і обравши необхідний пунк у випавшому меню. Окрім управління обліковими записами користувачів оператор сервера може подивитися срок дії їх ключів і знати коли вони перестануть бути дійсними.

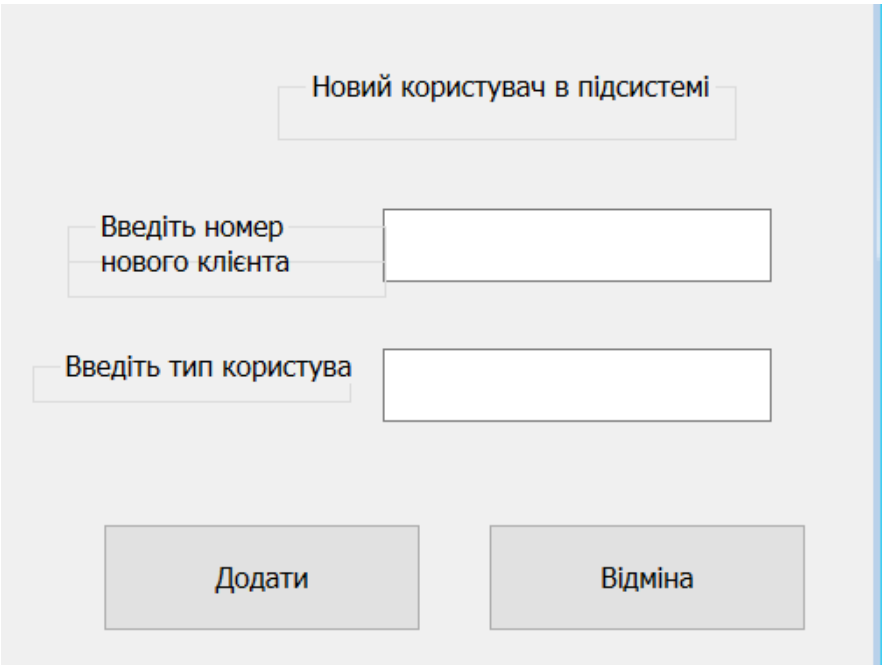

Рисунок 3.4 – Додавання нового користувача

Пройшовши авторизацію, користувач повинен створити пару ключів та надіслати публічний ключ на сервер. Отримавши публічний ключ сервер зберігає його у себе і використовує в процесі аутентифікації.

В нижній частині вікна інтерфейса оператора сервера знаходиться спикок користувачів даного сервера. Біля кожного з них відображаеться статус користувача в підсистемі. Всього існує три статуси:

- w(wait) – сервер готовий приймати нового користувача, але той ще не авторизувався в системі і не надіслав свій публічний ключ на сервер
- c(connect) – даний клієнт авторизован і активно працює в підсистемі
- d(disconnect) – клієнт тимчасово заблокован в звязку з багаторазовим непроходженням аутентифікації відправлених даних

Коли дані приходять на сервер вони видображаються у спеціальній зоні

інтерфейса у вигляді списку отриманих даних, імені відправника. Після перевірки даних біля кожного імені відображається успішність проходження аутенифікації.

Підсистема автоматично перевіряє ті дані, які на даний момент прислав користувач. Інші дані можна перевірити вручну вибравши: Верхне меню --> Файли --> Обрати дані, а потім після вибору необхідних даних натиснути Перевірити дані. Також для перевірки даних можно скористатися кнопками швидкого доступу для вибору та перевірки даних.

Для виводу перевірених даних конкретного користувача необхідно натиснути на його імя і на екрані зявиться список файлів даного користувача, які були перевірені.

У випадку коли дані не пройшли перевірку, про це повідомляється оператору сервера та системі, яка перемістить дані в “карантину папку” у якій він буде знаходитись до з'ясування причин не проходженням файлом аутентифікації.

Після перевірки всіх даних, підсистема відправляє користувачу звіт з результатами перевірки.

4.2 Реалізація клієнтської частини

На всіх клієнтських компьютерах системи встановлюється клієнтське ПЗ, розроблене для реалізації підсистеми аутентифікації.

Підсистема при роботі в обраній інформаційно-телекомунікаційній системі не потребує установки та налаштувань. Для початку роботи підсистеми необхідно просто запустити відповідне ПЗ і программа самостійно все встановить.

При першому запуску користувачу необхідно ввести свої дані для авторизації в підсистемі: логін та пароль. Форма авторизації така сама, як і оператора сервера. Далі підсистема робить запит до системи про отримання даних користувача.

Якщо в системі не зберігаються дані користувача чи по якійсь причині вони не можуть бути отримані, користувачу пропонується ввести їх у ручному режимі.

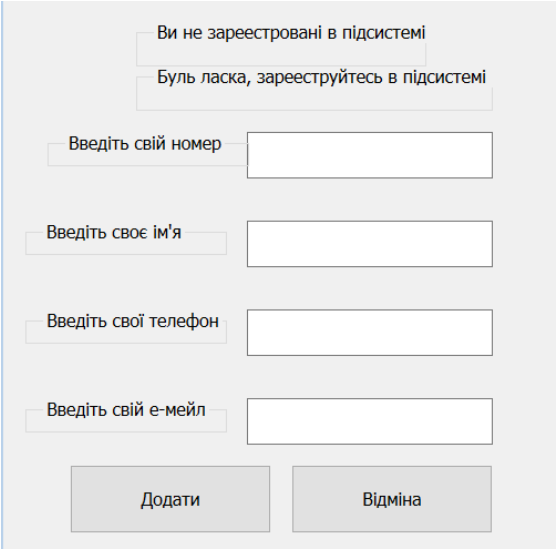

Рисунок 3.5 – Вікно реєстрації

Після успішної авторизації користувач отримує доступ до програмного інтерфейсу. Інтерфейс користувача дещо схожий на інтерфейс оператора сервера та має всі необхідні функції для успішної взаємодії користувача з підсистемою.

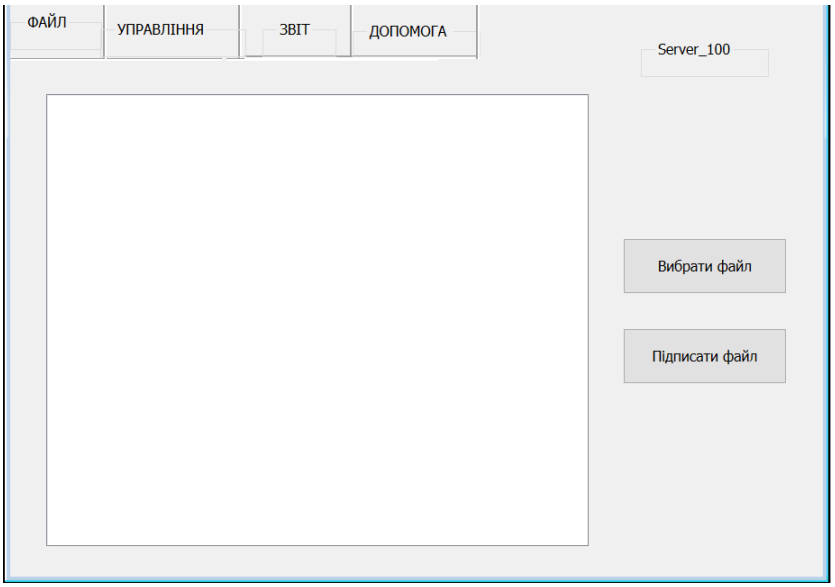

Рисунок 3.6 – Інтерфейс користувача

Після авторизації користувача в підсистемі необхідно створити пару ключів. Це можна зробити вибравши: Верхне меню --> Управління --> Створити пару ключів. Підсистема перевірить чи немає у цього користувача існуючої пари ключів, якщо ні – почне працювати алгоритм генерації публічного і приватного ключів. Ключі створюються на рік чи півроку, в залежності від типу користувача в системі.

В будь-який можен можна обновити пару ключів вибравши варіант в меню: Верхне меню --> Управління --> Обновити пару ключів.

За звичай ключі оновлюють у разі:

- втрати ключа
- підозри про заволодіння ключем третьою особою
- пошкодження ключа
- сплив термін дії ключа

Підсистема автоматично підписує дані, які користувач збирається відправляти на сервер. Якщо необхідно вручну підписати дані, то це можно зробити вибравши: Верхне меню --> Файли --> Обрати дані, а потім після вибору необхідних даних натиснути Підписати дані.

Також для підписання даних можно скористатися кнопками швидкого доступу для вибору та підписання даних.

4.3 Реалізація алгоритма ЕЦП

Удодатку Д зображена блок-схема генерації пари ключів та підписання нею даних користувача.

Для створення пари ключів в клієнтському ПЗ знаходиться программа, яка реалізує процес створення ключів по заданому алгоритму. Завдяки тому, що алгоритм знаходиться окремо від основної частини клієнтського ПЗ, це дає змогу легко модифікувати чи навіть замінити алгоритм генеарції пари ключів, у разі недостатньої стійкості поточного алгоритма при мінімальних змінах у підсистемі.

Алгоритм генерації пари ключів:

1. Генегуємо випадкове чило q розміром 160 біт за допомогою генератора випадкових чисел.
2. Перевіряємо на простоту число. Якщо це число просте, то воно береться в якості модуля для задання вектора. Якщо воно не просте, тоді генеруємо нове число, поки воно не буде простим.
3. Задаємо вектор G з розмірністю 4 і порядком q
4. Генеруємо секретний ключ x, який буде мати довжину у 254 біт. Це реалізується за допомогою пакета PyCrypto, що дозволяє згенерувати ключ за допомогою декількох команд.
5. Визначаємо вектор за формулою 4.1:

$$Y = G^x, \qquad (4.1)$$

де Y – відкритий ключ.

В результаті визначення вектора отримуємо публічний ключ де Y та серветний ключ X.

Після створення пари ключів вони будуть передані системі для відправки його по зашифрованому каналу звязку на сервер. Після отримання ключів сервером можна починати використовувати електронно-цифровий підпис для аутентифікації даних.

Алгоритм підписання даних:

1. Генеруємо випадкове число k
2. Визначаємо вектор за формулою 4.2:

$$R = G^k \qquad (4.2)$$

3. Об’єднуємо файл M, який буде передаватися на сервер та визначений вектор R в одне повідомлення
4. Визначаемо значення хеш-функції файла. Для отримання хеш-функції

файла використовується криптографычний алгоритм SHA-2

5. Обчислюємо другий елемент елетронно-цифрового підпису по формулі 4.3, який буде передаватися разом з хеш-функцією файла:

$$S = k + xE \bmod q \qquad (4.3)$$

6. Передаємо системі підписаний файл для відправки на сервер

Для роботи з векторами та виконання векторами та виконання матиематичних розрахунків використовується бібліотека NumPy, що дозволяе швидко виконати розрахунки за допомогою невеликої кількості команд.

У додатку Е зображена блок-схема генерації перевірки електронно-цифрового підпису. Після отримання файлів сервером її необхідно перевірити, щоб удостовіритись, що файли були створені користувачем і не були змінені третьою особою. Для цього викорстовується алгоритм перевірки підписа.

Алгоритм перевірка підпису:

1. Беремо публічний ключ користувача, дані якого будуть перевірені та визначаемо вектор за формулою 4.4:

$$R' = Y^{-E}G^{S} \qquad (4.4)$$

2. Обчислюємо значення хеш-функції отриманого файла
3. Порівнюємо значення хеш-функії вихідного файлу та хеш-функцію отриманого файлу. Якщо ці два значення рівні, то підпис для повідомлення є справжнім. В протилежному випадку підпис не приймається та про це повідомляється оператору сервера.

4.4 Розробка бази даних

Структура бази даних зображена в додатку Ж. База даних складається з 5 основних таблиць, які використовує сервер підсистеми аутентифікації даних

користувачів та ще додавткові таблиці, кількість яких залежить від кількості користувачів даного сервера.

Перша і найважливіша таблиця – це таблиця користувачів підсистеми. В даній таблиці зберігається список користувачів даного сервера.

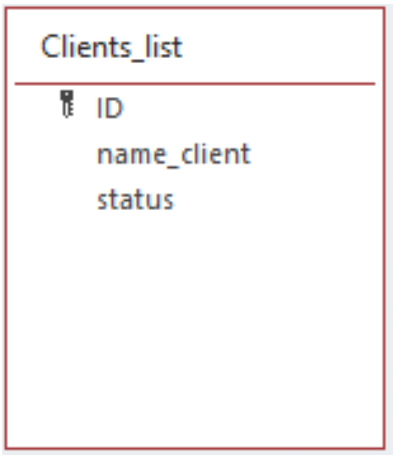

Рисунок – Опис таблиці “Список користувачів підсистеми”

name_client – імя користувача

status – статус користувача в підсистемі, може приймати три значення (wait, connect, disconnect)

В таблиці “Дані користувача” зберігаються усі необхідні дані про кожного користувача підсистеми.

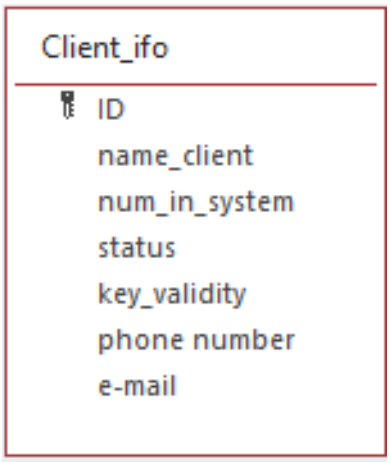

Рисунок – Опис таблиці “ Дані користувача”

name_client – імя користувача

num_in_system – номер комп’ютера в системі, для перевірки прав доступу обраного користувача до серверу

status – статус користувача в підсистемі, може приймати три значення (wait, connect, disconnect)

key_validly – відображає чи дійсний публічний ключ користувача

phone number – номер телефона користувача

e-mail – е-мейл користувача, на який будуть приходити повідомлення від серверу у разі проблем з аутентифікацією даних цього користувача

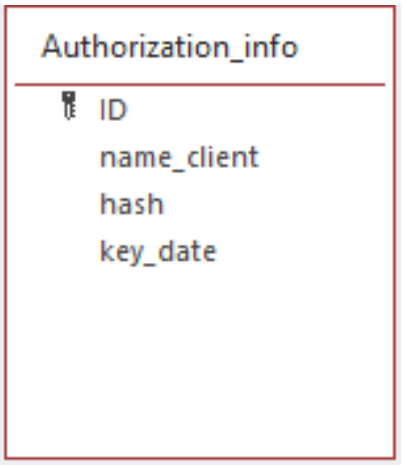

Рисунок – Опис таблиці “Дані для авторизації користувача”

name_client – імя користувача

hash – в даному полі містяться хеш-функція, яка використовується алгоритмом авторизації, для генерації та перевірки пароля користувача

key_date - дата, до якої ключі користувача дійсні

У таблиці “Контроль файлів” зберігаються дані про результат перевірки усіх отриманих від сервера файлів. На прикінці дня результат перевірки аналізуються та у вигляді статистичних даних відображаються у таблиці “Статистика”.

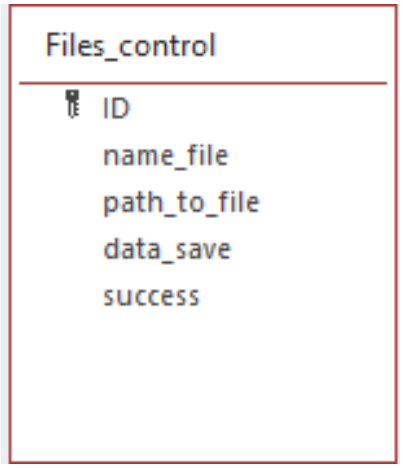

Рисунок – Опис таблиці “ Контроль файлів”

name_file – назва файлу, який проходив аутентифікацію

path_file – путь к директорії, в якій знаходиться даний файл

data_save – дата перевірки файла, отриманого від користувача

success – результат проходження аутентифікації

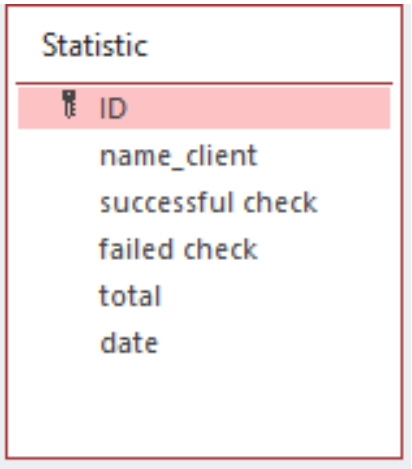

Рисунок – Опис таблиці “Статистика”

name_client – імя користувача

successful check – кількість файлів, які успішно пройшли аутентифікацію і були передані системі для подальшої роботи з ними

failed check – кількість файлів, які не пройшли аутентифікацію та були переміщені до “карантину” на час вирішення їх подальшої долі

total – загальна кількість перевірених файлів

date – дата перевірки

4.5 Діаграма розгортання підсистеми

Діаграма розгортання представлена в додатку К.

Перший вузол діаграми – сервер системи, за який відповідає оператор сервера. На серверах проводиться розгортання серверної частини підсистеми з інтерфейсом для оператора. Сервер відповідає за аутентифікацію даних користувачів. На серверній частині розгорнута база даних MySQL, призначена для зберігання даних о користувачах та дані результатів аутентифікації.

Сервер з’єднується з клієнтською частиною за допомогою TCP/IP.

Другий вузол – робоче місце користувача, представлене у вигляді персонального компьютера. На робочому місці користувача проводиться розгортання клієнтської частини з інтерфейсом користувача.

Висновок до розділу

В даному розділі було описано процес реалізації підсистеми аутентифікації даних в інформаційно-телекомунікаційній системі.

Розроблене програмне забезпечення, яке буде реалізовувати метод електронно-цифрового підпису складається з серверної та клієнтської частини.

Головна функція – сервера перевіряти отримані від користувача дані, а головна функція користувача – це підписувати свої дані перед відправкою їх на сервер.

С ервер і користувач може керувати підсистемою за допомогою командної строки чи за допомогою своїх інтерфейсів. Виконання адміністарторських задача більш зручніше робити за допомогою командної строки, адже для цього не потрібно бути фізично біля сервера.

Процес підписання та перевірки даних автоматизований і майже не потребує присутності користувача, окрім спірних випадків. Проте процес підписання та перевірки можно виконувати в ручному режимі, при якому буде зрубніше все робити через інтерфейс, де відображается увесь цей процес.

Також в підсистемі реалізована база даних. База даних зберігає дані про користувачів сервера та результати перевірки їхніх даних.

5 СТАРТАП-ПРОЕКТ

5.1. Опис ідеї проекту

Ідея даного стартап проекту полягає в створенні програмного продукту для реалізації аутентифікації даних користувача в автоматизованій інформаційно-телекомунакаційній системі. Унікальність продукта полягає у підвищеній криптостійкості електронного цифрового підпису зі збереженням швидкодії, а також універсальність даного підпису.

З погляду маркетингу продукт — це комплекс матеріальних і нематеріальних характеристик, які реалізовуються з метою задоволення потреб і забезпечення очікуваної вигоди як споживачем, так і виробником.

Для виробника у вигляді вигодою від продукту є отримання доходів та прибутків, або ж досягнення інших поставлених цілей у разі успішного продажу продукту.

Для споживачів продукт — це сукупність властивостей, які здатні задовольнити їхні потреби, розв'язувати певні конкретні проблеми і задачі.

Узагальнення цих ідей можна побачити в таблиці 5.1

Таблиця 5.1 Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигода для користувача
	1. Підсистема аутентифікації даних користувача в інформаційно-телекомунікаційній системі	Можливість аутентифікації даних користувача в системі. Генерація пари ключів для реалізації ЕЦП зі зменшеною довжиною ключів
	2. Підсистема аутентифікації даних користувача в автоматизованій	Можливість аутентифікації даних

	інформаційно-телекомунікаційній системі	користувача в атоматизованій системі. Генерація пари ключів для реалізації ЕЦП зі зменшеною довжиною ключів Автоматизовація процесу підписання та перевірки даних користувача

Основними конкурентами даного проекту є системи аутентифікаці. На даний момент таких систем небогато, і більшість з них має значні недоліки. Порівняння даного проекту такими система представлено в таблиці 5.2

Таблиця 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї старап проекту

№	Техніко-економічні характеристики ідеї	(Потенційні) товари/концепції конкурентів			Слабка сторона	Нейтральна сторона	Сильна сторона
		Мій проект	Крипто ПРО	Крипто АРМ			
1	Створення пари ключів	Присутнє	Присутнє	Присутнє	-	+	-
2	Використання нових задач	Присутнє	Присутнє	Відсутнє	-	-	+
3	Робота в автоматизованій системі	Присутнє	Присутнє	Відсутнє	-	-	+

Продовження таблиці 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї старап проекту

4	Незалежність від мережі інтернет	Присутнє	Відсутнє	Присутнє	-	+	-
5	Можливість працювати в різних ОС	Присутнє	Присутнє	Присутнє	-	-	+
6	Контроль автентичності даних	Присутнє	Присутнє	Присутнє	-	-	+

5.2 Технологічний аудит ідеї проекту

Оскільки проект вирішує конкретну програмну проблему, то більшість технологій для реалізації даного проекту є доступною для розробки. Технології, що доцільно використовувати в рамкках даного проекту можна побачити в таблиці 5.3.

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології її реалізації	Наявність технології	Доступність технології
1	Процедури реалізації ЕЦП	Мови загального програмування	Python	Вільні для використання
2	Інтерфей коритувача	Мова та середовище программування	Python, PythonQt	Вільні для використання
Обрана технологія реалізації ідеї проекту: Скриптова/інтерпретована (Python)				

5.3 Аналіз ринкових можливостей стартап проекту

Аналіз ринкових можливостей є однією з початкових та важливих стадій встановлення ринкових можливостей. На початковому етапі розробки стартап-проекту необхідно реально оцінити свої можливості. Це особливо значимо, оскільки жодна із діючих на ринку структур не може постійно покладатися на досягнуте місце на ринку, адже, цілком зрозуміло, що тривалість життя послуги чи товару незначна.

Кожна компанія повинна реально оцінити свої потенційні можливості, зокрема, сильні і слабкі сторони своєї діяльності.

Під час аналізу ринкових можливостей необхідно врахувати такі складові:

- систему маркетингових досліджень і маркетингової інформації;
- оцінку маркетингового середовища;
- аналіз ринків індивідуальних споживачів;
- аналіз ринку підприємства.

Аналіз ринкових можливостей включає детальне вивчення:

- соціально-політичного стану середовища;
- статистики ринків країни;
- конкуренції на існуючому ринку;
- особливостей поведінки споживачів;
- особливостей умов просування і збуту;
- стандартизації та вимог якості.

Попередня характеристика потенційного ринку стартап-проекту представлена в таблиці 5.4.

Таблиця 5.4 – Попередня характеристика потенційного ринку стартаппроекту

№	Показники стану ринку (найменування)	Характеристики
1	Кількість головних гравців, од	5
2	Загальний обсяг продаж, грн/ум.од	100000 грн/ум.од

Продовження таблиці 5.4 – Попередня характеристика потенційного ринку стартаппроекту

3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Недискримінаційні якісні
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	70%

Характеристика потенційних клієнтів стартап-проекту представлена в таблиці 5.5

Таблиця 5.5 – Характеристика потенційних клієнтів стартап-проекту

№	Проблема, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Алгоритм ЕЦП з короткою довжиною ключів	1. Особисте користування 2. Малий та середній бізнес	Наявність вимог до довжини ключів, критостійкості	Зменшення часу створення ЕЦП зі збереженням рівня криптостійкості, легкість в налаштуванні

Продовження таблиці 5.5 – Характеристика потенційних клієнтів стартап-проекту

2	Автоматизація процеса аутентифікації	1. Малий та середній бізнес	Різне апаратне забезпечення, кваліфкація персоналу	Зменшення впливу людського фактору

Зовнішній аналіз - це процес оцінки зовнішніх факторів, які об’єктивно існують в середовищі функціонування проекут, на які вона безпосередньо не може впливати.

Цілі зовнішнього анілізу:

- визначити можливості та загрози;
- ідентифікувати ключові фактори успіху в обраній сфері бізнесу.

Сприятливі можливості - це фактори зовнішнього середовища, які допомагають досягненню цілей проекту.

Загрози - це зовнішні умови, які обмежують можливості фірми досягти мети. Одні й ті ж самі фактори можуть створювати як додаткові можливості, так і додаткові перешкоди.

Фактори загроз, що перешкоджають впровадженю проекта представлені в таблиці 5.6.

Таблиця 5.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Крадіжка інтелектуальної власності	Крадіжка ідеї або ключової інтелектуальної інновації	Відсудження прав інтелектуальної власності. Попередження користувачів із подальшою співпрацею для мінімізації фактор загрози

Продовження таблиці 5.6 – Фактори загроз

2	Отримання несанкціонованого доступу сторонніми особами	Хакерська атака що може призвести до компроментації даних клієнтів	Залучення спеціалістів з інформаційної безпеки Використання засобів шифрування
3	Відсутність ринку	Відсутність шляху збуту товару внаслідок помилкового орієнтування	Ретельний огляд проблем потеційних клієнтів

Фактори можливостей, які сприяють впровадженю проекта представлені в таблиці 5.7

Таблиця 5.7 - Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Успішна маркетингова політика	Отримання капіталу що необхідний для реалізації продукту	Розробка продукту
2	Успішна маркетингова політика	В результаті проведеної маркетингової політики отримана висока зацікавленість користувачів	Підтримка стабільної роботи підсистеми та проведення масштабування системи Збільшення цін на використання сервісу Використання подібної маркетингової стратегії надалі для залучення нових користувачів

Продовження таблиці 5.7 - Фактори можливостей

З	Поглинання конкурентами	Пропозиція купівлі проекту або розроблених технологій одним із конкурентів	Розвиток розроблених технологій. Оцінка вартості розроблених технологій

Наступний етап аналізу мікросередовища - це аналіз конкурентів. Конкурентний аналіз спрямований на визначення можливостей, загроз і відшукування стратегічних невизначеностей, що можуть створюватися конкурентами, що суперничають на певному ринку. Аналіз починається з визначення головних і потенційних конкурентів. Потім переходять до більш глибокого і ретельного вивчення різних аспектів їхньої діяльності: місії, цілей, стратегій, сильних і слабких сторін.

Для проведення успішної рекламної компінії необхідно виділити основні переваги підсистеми в порівняні з іншими підсистемами:

- автоматизація та прискорення повсякденних задач
- висока криптостійкість підсистеми
- зменшена довжина ключів, які використовуються для реалізації електронно-цифрового підпису
- перевірка сроків дії ключів та нагадування користувачу про спилвання сроку дії ключа
- постійний контроль даних користувачів з веденням статистики в базі даних
- скорочення витрат часу на підписання та перевірки даних
- зменшення помилок в роботі
- працювати можна, як через консоль, так і через візуальний інтерфейс

Ступеневий аналіз конкуренції на ринку та аналіз конкуренції в галузі за М. Портером представлені в таблиці 5.8 та таблиці 5.9 відповідно.

Таблиця 5.8 - Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Олігополія	Незначна кількість конкурентів; Схожість технологій, які використовуються	Інформування ринку щодо появи нової платформи
Галузевий	Загроза появи нових конкурентів; Висока потреба у товарі	Інформування ринку щодо якості використовування новаторської технології; Пропозиція гнучких цін
Внутрішньогалузева	Діяльність в одній галузі економіки; Надання сервісів одного типу	Зменшення вартості сервісу; Примноження каналів розподілу
Товарно-видова	Надання різних сервісів одного типу	Маркетингова політика
Цінова	Використання цін для покращення економічних умов збуту	Зменшення вартості платформи; Використання нових каналів розподілу
Марочна	Пропозиція схожої платформи; Спільна цільова аудиторія	Інформування ринку щодо появи нової платформи

Стан конкуренції в галузі залежить від п'яти основних конкурентних сил

(модель п'яти конкурентних сил, розроблена професором Гарвардської школи бізнесу М. Портером:

1. Суперництво між продавцями усередині галузі.
2. Фірми, що пропонують товари-замінники (субститути).
3. Можливість появи нових конкурентів усередині галузі.
4. Здатність постачальників сировини, матеріалів і комплектуючих, які використовуються фірмою, диктувати свої умови. І
5. Здатність споживачів продукції фірми диктувати свої умови.

Дана модель дає змогу визначити найкращу відповідність між внутрішнім станом організації і дією сил у її зовнішньому оточенні.

Суперництво між продавцями усередині галузі. Конкуренція між двома та більше продавцями, що пропонують однотипні товари і послуги, виникає у зв'язку з тим, що в однієї чи декількох фірм з'являється можливість краще задовольнити потреби споживача.

Основні засоби конкурентної боротьби:

- Низька ціна
- Покращення характеристик товару
- Високий рівень обслуговування
- Постійне оновлення товарного ассортименту
- Використання слабких сторін конкурентів

Не зважаючі на наявність та рівень конкуренції потрібно розробляти ефективну стратегію, що забезпечить перевагу над конкурентами. В будь який час може прийти продавець з іншої галузі, який буде мати значні ресурси, добре підготовлені виробничі потужності. Маючи бажання закріпитись на даному ринку, він стане потенційним конкурентом.

Застосування для аналізу конкуренції в галузі моделі п'яти конкурентних сил М. Портера дозволяє визначити структуру цих сил, оцінити кожну силу і приступити до формування конкурентної стратегії. У таблиці 5.9 представлений аналіз конкуренції в галузі за моделею М.Портера

Таблиця 5.9 - Аналіз конкуренції в галузі за М. Портером

Складові аналізу		Висновки
Прямі конкуренти в галузі	КРИТОПРО, КриптоАРМ CryptoLibV2	Середній ринок конкуренції
Потенційні конкуренти	Розмір капіталовкладень, забезпечення гнучких цін,	Можливості входу на ринок забезпечить мінімізація цін, швидкість та простота надавання послуги споживачам і співпраця із головними гравцями ринку. В результаті аналізу проектів на інтернет-платформах потенційних конкурентів знайдено не було
Постачальники	Відсутні	Відсутні
Клієнти	Змінні витрати: Виробничі непрямі дегресивні - Системи інформації: пропаганда, реклама та директ-маркетинг, - Рівень чутливості до цін: споживачі орієнтовані на цінність продукту -Продуктова диференціація: якість, спосіб отримання сервісу, швидкість обслуговування	Клієнти диктують умови гнучкості цінової політики, високої і довгострокової якості послуг та наявність кооперації із сервісами, що вони використовують

Продовження таблиці 5.9 - Аналіз конкуренції в галузі за М. Портером

Товари-замінники	Копіювання функціоналу, Монополізація дистриб'юторів, Демпінгування	Пропонування вигідних умов дистриб'юторам, забезпечення захисту інтелектуальної власності, гнучкість цінової політики

На основі налізу конкуренції, а також із урахуванням характеристик ідеї проекту, вимог споживачів до товару та факторів маркетингового середовища визначається та обґрунтовується перелік факторів конкурентоспроможності. В таблиці 5.10 предствлений аналіз факторів конкурентоспроможності.

Таблиця 5.10 - Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1.	Унікальність сервісу	Розроблений продукт має унікальні пропозиції: Реалізація ЕЦП з короткою довжиною ключів
2.	Цінова політика	Отримання прибутку здійснюється за рахунок гнучкої моделі оплати
3.	Модель "бізнес для бізнесу"	Бізнес модель грунтується на унікальності пропозиції і співпраці з власниками веб ресурсів. Даний підхід дозволить обійти цінову конкуренцію на ринку цільової аудиторії

SWOT-аналіз – це метод стратегічного планування, що полягає у виявленні факторів внутрішнього і зовнішнього середовища організації і поділі їх на

чотири категорії: сильні сторони, слабкі сторони, загрози та можливості.
Оцінка місця підприємства в галузі необхідна для одержання попередньої всебічної оцінки стратегічного положення підприємства, а також для розробки переліку стратегічних дій.

SWOT – аналіз має декулька етапів, які виконуються у певній послідовності.

На першому етапі вивчається зовнішнє середовище підприемства та виділення факторі, які сприяють розвитку підприємства та факторів, які несуть загрозу.

На другому етапі проводиться аніліз внутрішнього середовища підприємства, що дозволяє визначити його сильні й слабкі сторони.

Таблиця SWOT- аналізу відображає тісний зв'язок з часом проведення аналізу оскільки фактори зовнішнього й внутрішнього середовища піддаються змінам.

На третьому етапі зіставляються зовнішні можливості та загрози із внутрішнім потенціалом і обмеженнями, що дозволяє визначити здатність даного підприємства скористатися наявними ринковими можливостями й мінімізувати негативний вплив зовнішніх загроз. SWOT – аналіз стартап-проекту представлений у таблиці 5.11.

Таблиця 5.11 - SWOT - аналіз стартап-проекту

Сильні сторони: Унікальність пропозиції; низькі ціни	Слабкі сторони: Нестача капіталовкладень; відсутність можливості використання смарт-карт
Можливості: Інвестиції; висока зацікавленість цільової аудиторії	Загрози: Крадіжка інтелектуальної власності компроментація даних клієнтів відсутність ринку

Результатом проведення SWOT-аналізу є система можливих стратегічних дій, спрямованих на посилення конкурентних позицій підприємства і його розвиток. Альтернативи ринкового впровадження стартап-проекту представлені в таблиці 5.12.

Таблиця 5.12. Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Розшерення можливостей сервісу	Ймовірне	5 місяців
2	Додання нових бізнес моделей	Малоймовірне	10 місяців
3	Пошук бізнесів інших галузей для співпраці	Малоймові рне	5 місяців
Обрана альтернатива: Розшерення можливостей сервісу			

5.4 Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначеення стратегії охоплення ринку: опис цільових груп потенційних споживачів.

Підприємство може вибрати один чи кілька сегментів ринку залежно від вибраної стратегії охоплення ринку.

Існує три стратегії охоплення ринку: масовий маркетинг, цільовий маркетинт та дифиринційований маркетинг.

За результатами аналізу цільових груп потенційних споживачів, який представлений у таблиці 5.13 буде обрано статегію охоплення ринку.

Таблиця 5.13. Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Користувачі персональних комп'ютерів	Висока	65%	Середня	Низькі бар’єри входу
2	ІТ-підрозділи середнього бізнесу	Мала	42%	Середня	Високі бар’єри входу
3	Власники	Висока	76%	Середня	Низькі бар’єри входу

У разі використання стратегії масового маркетингу підприємство орієнтується на широкий ринок споживачів з використанням одного базового комплексу маркетингу, сегментацію ринку не проводять.

Цю стратегію доцільно застосовувати на однорідному ринку, де всі споживачі виявляють інтерес до одного товару й однаково реагують на запропоновані маркетингові заходи

Цільовий чи концентрований маркетинг передбачає орієнтацію на вузьку специфічну групу споживачів через спеціалізований комплекс маркетингу, спрямований на задоволення потреб саме цього сегмента.

Стратегія цільового маркетингу ефективна насамперед для невеликих чи спеціалізованих підприємств, які виробляють продукцію конкретного призначення в обмеженій кількості.

Диференційований маркетинг передбачає охоплення декількох сегментів ринку і розроблення для кожного з них окремого комплексу маркетингу. Головна перевага полягає у тому, що для фірми зменшується рівень ризику і негативні економічні наслідки у разі невдачі на якомусь сегменті. Компанія краще задовольняє інтереси цільових споживачів

Вибір базової стратегії розвитку представлений у таблиці 5.14.

Таблиця 5.14 - Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Надання платформи малому та середньому бізнесу	Вибірковий розподіл	Здатність протистояти прямим конкурентам; низькі витрати; ефективна співпраця	Стратегія диференцації

Стратегія конкуренції - це наступальні чи оборонні дії підприємств, спрямовані на досягнення стійкого становища в галузі, з метою успішного подолання п'яти чинників конкуренції і гарантування максимальної віддачі від капіталовкладень

Існує три основні різновиди стратегії конкурентної поведінки:

- наступальна;
- оборонна;
- кооперацийна

Конкурентна перевага майже завжди досягається за рахунок

наступальних стратегічних дій підприємства.

Визначення базової стратегії конкурентної поведінки представлене у таблиці 5.15

Таблиця 5.15. Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першопрохідце» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента	Стратегія конкуреноїповедінки*
1	Ні	Забирати та залучати нових	Десктопний інтерфейс керування	Стратегія лідера. Розширення первинного попиту

Основна мета стратегії позиціонування полягає в тому, щоб виробити прихильність споживача до товару фірми через визначення позитивних відмінностей цього товару від товарів конкурентів.

Існують такі стратегії позиціонування:

- позиціонування на показниках якості;
- позиціонування за співвідношенням “ціна - якість”. Суть цієї стратегії полягає в знаходженні оптимального поєднання цих показників і доведення його до споживача;
- позиціонування на основі порівняння товару фірми з товарами конкурентів. Ця стратегія реалізується в порівняльній рекламі;
- позиціонування за сферою застосування;
- позиціонування за відмінними особливостями споживача, якому

пропонується товар;

- позиціонування за різновидом товару, який пропонується у продаж;
- позиціонування на низькій ціні. Ця стратегія застосовується багатьма фірмами, які діють у різних сферах бізнесу;
- позиціонування на сервісному обслуговування;
- позиціонування на позитивних особливостях технології;
- позиціонування на іміджі. Ця стратегія спирається на вже набуту фірмою репутацію серед споживачів.

Фірма, здійснюючи стратегію позиціонування, може обирати одну, дві або три ознаки.

Дослідження свідчать, що якщо позиціонування здійснюється більше, ніж за трьома ознаками, то воно є неефективним, оскільки не відкладається у свідомості споживача.

Визначення стратегії позиціонування представлене у таблиці 5.16

Таблиця 5.16. Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
Відповідність затвердженим характеристикам; висока ступінь надійності системи простий інтерфейс	Стратегія диференціації	Формування регулярного попиту	Інноваційність технології; низькі ціни; простота використання

5.5 Розроблення маркетингової програми стартап-проекту

Традиційно склалося, що у процесі інноваційної діяльності розробляють та перевіряють концепцію самого нового товару.

Тобто, вважають, що, якщо ідея прийнята до реалізації, то концепцію самої ідеї перевіряти не має сенсу. При цьому під ідеєю розуміється загальний опис товару, а під концепцією - ідея, яка розроблена і сформульована з точки зору значимих для споживача характеристик товару. У зв'язку з тим, що покупець купує не ідею, а концепцію, саму ідею необхідно перетворити в декілька альтернативних концепцій. Потім виявити ступінь привабливості кожної концепції і обрати серед них найкращу.

Для цього необхідно застосовувати дві концептуальні вимоги: скорочувати термін між висуванням ідеї і виходом нового товару на ринок. Під час вироблення і перевірки концепції головну увагу необхідно приділяти не виробничим проблемам, а прогнозуванню попиту.

Таблиця 5.17. Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода,яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Створення підсистеми аутентифікації даних	Реалізація аутентифікації даних на базі ЕЦП	Якість надання послуг та простота використання

Товар — це продукт праці, зроблений для продажу, має споживчу цінність для покупця, тобто здатність задовольнити потребу.

Три рівні товару

1. Товар за задумом — це подання цільової функції, базової вигоди, заради якої купується товар.
2. Товар у реальному виконанні включає в себе якість, властивості, дизайн, упаковку, ціну.
3. Товар з підкріпленням (супроводом) — додаткові послуги та переваги для споживача, що створюються на основі товару за задумом і товару в реальному виконанні (гарантії якості , доставка, умови оплати та ін)

Таблиця 5.18. Опис трьох рівнів моделі товару

<table>
<tr><td>Рівні товару</td><td colspan="3">Сутність та складові</td></tr>
<tr><td>I. Товар за задумом</td><td colspan="3">Програмний продукт що надає можливість облегшити аутентифікацію і збільшити криптостійкість</td></tr>
<tr><td rowspan="5"></td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td>Кількість</td><td></td><td>1 шт.</td></tr>
<tr><td colspan="3">Якість: стандарти якості постачання програмних продуктів</td></tr>
<tr><td colspan="3">Пакування: Завантеження з інтернет ресурсу</td></tr>
<tr><td colspan="3">Марка: AutoECP</td></tr>
<tr><td rowspan="2"></td><td colspan="3">Програмний продукт</td></tr>
<tr><td colspan="3">Програмний продукт, технічна підтримка та підписка на оновлення</td></tr>
<tr><td colspan="4">За рахунок чого потенційний товар буде захищено від копіювання: захист інтелектуальної власності</td></tr>
</table>

Визначення меж встановлення ціни складаєтьсяз 7 етапів:

На первшому етапі визначається попит на товар, його обсягу й динаміку. Також визначается можливості покупця придбати даний товар за

запропонованою ціною. Так, деякі фірми пропонують покупцям нового товару самим назначити йому ціну і пропонують даний товар за цією ціною на ринку, що підвищує престиж фірми у споживачів і слугує непоганою рекламою.

На другому етапі обирається вигідна ціна для підприємства. Дана ціна при множені на обсяг продаж повина забезпечувати максимальний рівень прибутку. Ціна визначена на даному етапі повина бути максимально вигідною для підприемства.

На третьому етапі визначается рівень ціни товару та її структури на основі порівняння з аналогічними товарами фірм конкурентів, здійснюючи при цьому коректування ціни за техніко-економічними параметрами якості та іншими складовими конкурентоспроможності. Таке порівняння передбачає закупівлю товару у конкурентів, використання ціни прейскурантів, опитування покупців.

Також на даному етапі відбувається приведення ціни до единих умов: терміну поставки, умов і валюти платежу та коректування ціни з урахуванням можливої реакції на неї конкурентів. Ігнорування або недостатнє врахування цієї обставини може призвести до зниження ефективності формування цінової політики фірми та її підприємницької діяльності.

На четвертому етапі визначается верхня і нижня межі – порог ціни, а також можливих меж і умов зниження цін. Також визначається динаміка цін в залежності від від стадії життєвого циклу товару.

На п’ятому етапі визначается співвідношення цін між товарами та їх модифікаціями. Беруться до уваги різниці в собівартості, в оцінках даних товарів споживачами, ціни конкурентів. За умов великого розриву в цінах між двома аналогічними товарами споживач купує більш досконалий товар, а у випадку відсутності цінової різниці – менш досконалий.

На даному етапі визначаются цінові ліній, пов'язані з продажем товарів у діапазоні цін, де кожна ціна відбиває рівень якості різних моделей одного й того ж виду товарів. Також визначается ціна на додаткові й допоміжні товари. Відбувається установлення ціни на обов'язкові речі – аксесуари, які

доповнюють основні товари. У результаті утворюється складна сітка цін.

На шостому етапі ведеться розробка тактик цін та розрахунок усіх можливих варіантів знижок-надбавок. Знижки використовують для того, щоб реагувати на більш низькі ціни конкурентів, скоротити запаси, ліквідувати залишки товару.

На сьомому етапі визначается контрактна ціна, за якою товар може бути проданий та визначаеться експортна ціна. Також визначаются базові умови цін, які враховують не тільки вартість самого товару, а й обов'язки продавця за його транспортуванням, страхуванням.

Визначені межі встановлення цін наведені у таблиці 5.19

Таблиця 5.19. Визначення меж встановлення ціни

Рівень цін на товари- замінники	Рівень цін на товари- аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
3 – 4 usd./міс.	5 – 10 usd./міс.	10 000 грн./міс.	2 – 3 usd./міс.

Для забезпечення ефективної реалізації вироблених товарів підприємства здійснюють комплекс заходів, який забезпечує фізичне переміщення та розподіл товарної маси у ринковому просторі, доведення товарів до споживачів і організацію їх ефективного споживання або використання. Все це присутнє у розробці маркетингової збутової стратегії.

Слід враховувати, що збут - це один а головних елементів маркетингу, який стоїть позаду таких елементів, як виявлення споживчих потреб, розробка товарів і встановлення на них відповідної ціни, налагодження системи їх ефективного стимулювання.

Роль збуту у маркетинговій діяльності обумовлена наступними

обставинами:

1. у сфері збуту визначається результат комерційного виробництва;
2. пристосування збутової мережі до запитів споживачів впливає на перемогу у конкурентній боротьбі;
3. збутова мережа продовжує процес виробництва, беручи на себе функцію доробки товарів, сортування, розфасовку і упакування;
4. на стадії збуту чітко вимальовуються смаки, запити і переваги споживачів. Дослідження основних форм і методів збуту спрямоване на пошук перспективних засобів просування товарів від виробника до кінцевого споживача і організацію роздрібної торгівлі на основі все стороннього аналізу і оцінки ефективності використовуваних каналів і способів розподілу і збуту.

Основними критеріями відбору каналів збуту є:

- швидкість товаропросування;
- рівень витрат обігу;
- обсяги реалізації товарів.

У таблиці 5.20 представлено визначення формування збуту.

Таблиця 5.20. Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Закупівля здійснюється через довірені джерела	Інформування користувачів Доступ користувачів до сервісу	Канал одного рівня	Селективна з використаннм комбінованого

Маркетингові комунікації - це засоби, за допомогою яких фірми намагаються інформувати, переконувати і нагадувати споживачам,

безпосередньо або побічно, про свої товари і торгових марках.

Управління маркетинговими комунікаціями - цілеспрямована діяльність компанії з регулювання ринкової стійкості за допомогою інформаційних технологій.

Маркетингові комунікації виконують цілий ряд функцій, що дозволяє споживачам бути поінформованими про товари і послуги, а виробникам про потреби споживачів. Сьогодні діяльність окремих людей, груп і організацій безпосередньо залежить від їх інформованості і здатності ефективно використовувати наявну інформацію.

В інформаційному суспільстві зміняться не тільки виробництво, але і весь устрій життя, система цінностей. Основна відмінність сучасного ринку полягає в тому, що інформація і знання рухаються в обох напрямках: від продавця до споживача і від споживача до продавця.

Розвиток і поширення нових технологій, тенденції глобалізації та інформатизації, збільшення кількості ринкових альтернатив зумовили перехід суспільства від індустріального типу розвитку до інформаційного. У життєдіяльності сучасного суспільства, сучасної економіки все більшої значущості отримують інформація, системи та технології її збору, аналізу та впливу на аудиторію.

Таблиця 5.21. Концепція маркетингових комунікацій

Специфіка поведінки цільових клієнтів	Автоматизація бізнес-процесів Вимоги до високодоступності та відмовостійкості
Канали комунікацій, якими користуються цільові клієнти	Прямі офіційні
Ключові позиції, обрані для позиціонування	Послідовність в реалізації обраної позиції Доступність та об’єктивність інформації про фірму і товар Унікальність послуги

Завдання рекламного повідомлення	Формування у цільової аудиторії обізнаності про появу нового продукту Інформування користувачів про властивості та переваги продукту Інформування користувачів про нові способи використання відомого продукту Пояснення цільовій аудиторії принципу роботи платформи Виправити у користувачів неправильніпредставлення про продукт
Концепція рекламного звернення	Раціоналістична стратегія реклами

Висновок до розділу

В даному розділі було розроблено стартап проекту.

В ході розробки стартап-проекту була описана ідея проекту, визначені слабкі та сильні сторони в порівнянні з схожими продуктами, проведені маркетингові дослідження

В ході маркетингового дослідження були виявлені попередні характеристики потенційного ринку, можливі фактори загроз та ризикуів.

Було проаналізовано конкуренцію на ринку схожих товарів, який є достатньо великим.

Також була розроблена базова стратегія розвитку проекту.

Розроблена система підходить для використання на малих підприємствах.

Вона дозволяє зменшити витрачений на тривіальну роботу час і збільшити продуктивність підприємства, що веде до збільшення доходів підприємства.

Також було розроблено маркетингову стратегію продукту, унікальну торгову пропозицію.

Завдяки напрцюванням створеним у цьому розділі можна запускати повноцінну рекламну кампанію.

ВИСНОВОК

В даній магістерській дисертації була розроблена підсистема аутентифікаї даних користувача в інформаційно-телекомунікаційній сисстемі.

Для розв’язання даної проблеми були використані методи електронно-цифрового підпису. Після проведення аналізу існуючих рішень було виявлено ряд недоліків, який які не дозволяють використовувати дані рішення в обраній інформаційно-телекомунікаційній системі. Тому, було прийнято рішення розробити свою підсистему аутентифікації даних.

Для зменшення довжини ключів без втрати криптостійкості було досліджено можливість використання некомутативних групп. Було досліджено кінцеві групи кілець, способи їх задання. Під час дослідження було виявлено, що даний алгоритм має великий недолік – низьку продуктивність. Щоб підвищити продуктивність алгоритму було викоористано метод паралельного обчислення векторів.

На основі всіх отриманих результатів досліджень були сформовані вимоги до системи. Далі були розроблені діаграма використання системи, функціональна та структурні схеми, які надають більш глибоке представлення роботи системи. Далі був приведений обґрунтований вибір вибраних технологій для реалізації системи.

Далі була проведена розробка алгоритма створення ключів, перевірки та підписання даних електронно-цифровим підписом. Було розроблено підсистему аутентифікації даних користувачів з використанням розробленого алгоритму. Далі були розроблені діаграми розгортань та послідовності для пояснення роботи системи.

На основі зроблених напрацювань був розроблений стартап проект.

ЛІТЕРАТУРА

1. Schneier, Bruce. Applied Cryptography, John Wiley & Sons, 1994
2. Daniel R. L. Brown. Generic Groups, Collision Resistance, and ECDSA / Daniel R. L. Brown., 2002.
3. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System / Nakamoto., 2008.
4. Винберг Э. Б. Курс алгебры. — новое изд., перераб. и доп. — М. : МЦНМО, 2011. — 592 с. — 2000 прим.
5. Schoof R. Nonsingular plane cubic curves over finite fields / René Schoof // Nonsingular plane cubic curves over finite fields / René Schoof., 1987. – (A). – С. 183–211.
6. Захист інформації в інформаційно-телекомунікаційних системах : Навч. посіб. для студ. Ч. 1. Криптографічний захист інформації / І.Д. Горбенко, Т.О. Гріненко; Харк. нац. ун-т радіоелектрон. - Х., 2004. - 368 с. - Бібліогр.: 73 назв. - укр.
7. Моделі і системи оцінювання, обробки та захисту фінансової інформації: Моногр. / Г.М. Азаренкова, С.В. Гадецька, І.Д. Горбенко, Ю.В. Дубницький, О.О. Єгоршин; Ред.: О.В. Васюренко. - Х.: Константа, 2005. - 380 с. - Бібліогр.: с. 368-380. - укр.
8. UML diagrams [Електронний ресурс] – Режим доступу до ресурсу: https://www.uml-diagrams.org/.
9. Python [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Python_(programming_language).